

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TASARIMCI BAKIŞ AÇISIYLA
HESAPLAMALI DÜŞÜNMEDE PROSEDÜREL SOYUTLAMANIN
DEĞERLENDİRİLMESİ İÇİN BİR YÖNTEM**

DOKTORA TEZİ

Elif Belkıs ÖKSÜZ

Mimarlık Anabilim Dalı

Mimari Tasarım Programı

NİSAN 2020

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TASARIMCI BAKIŞ AÇISIYLA
HESAPLAMALI DÜŞÜNMEDE PROSEDÜREL SOYUTLAMANIN
DEĞERLENDİRİLMESİ İÇİN BİR YÖNTEM**

DOKTORA TEZİ

**Elif Belkıs ÖKSÜZ
(502132013)**

Mimarlık Anabilim Dalı

Mimari Tasarım Programı

Tez Danışmanı: Prof. Dr. Gülen ÇAĞDAŞ

NİSAN 2020

İTÜ, Fen Bilimleri Enstitüsü'nün 502132013 numaralı Doktora Öğrencisi Elif Belkıs ÖKSÜZ, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “TASARIMCI BAKIŞ AÇISIYLA HESAPLAMALI DÜŞÜNMEDE PROSEDÜREL SOYUTLAMANIN DEĞERLENDİRİLMESİ İÇİN BİR YÖNTEM” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Gülen ÇAĞDAŞ**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. Dr. Leman Figen GÜL**
İstanbul Teknik Üniversitesi

Öğr. Gör. Dr. Bahar BEŞLİOĞLU
Mimar Sinan Güzel Sanatlar Üniversitesi

Öğr. Gör. Dr. Bahadır NUMAN
İstanbul Teknik Üniversitesi

Dr. Öğr.Üye. Güven ÇATAK
Bahçeşehir Üniversitesi

Teslim Tarihi : **16 Mart 2020**
Savunma Tarihi : **03 Nisan 2020**





evrene,



ÖNSÖZ

Tezi bitirene kadar hayatıma girmiş ve hayatımdan çıkmış herkese katkılarından dolayı teşekkür ederim.

Nisan 2020

Elif Belkıs Öksüz
(Mimar)





İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
ÖZET.....	xvii
SUMMARY	xix
1. GİRİŞ	1
1.1 Problemin Tanımı ve Tezin Amacı	1
1.2 Araştırma Yöntemi ve İçerik	3
1.3 Araştırmanın Özgün Değeri ve Literatüre Katkısı	4
2. HESAPLAMALI DÜŞÜNMENİN MİMARLIK EĞİTİMİNDEKİ ÖĞRETİM KÜLTÜRÜ VE PEDAGOJİK UYUMU.....	7
2.1 Yirmi Birinci Yüzyıl Becerisi Olarak Hesaplamalı Düşünme	7
2.2 Yeni Başlayanlar için Hesaplamalı Düşünmenin Öğretim Kültürü	10
2.3 Mimarlık Eğitiminde Hesaplamalı Düşünme ve Pedagojik Uyumu	18
3. GÖRSEL BİR DÜŞÜNÜRÜN HESAPLAMALI DÜŞÜNME YAKLAŞIMI	25
3.1 Görsel Düşünürler İçin Temel Tasarım Kavramları: Soyutlama, Örüntü Oluşturma ve Ayırıştırma	26
3.2 Tasarımcı Bakış Açısıyla Problem Formüle Etme: Algoritmik Düşünme.....	30
4. TASARIMCI BAKIŞ AÇISI İLE HESAPLAMALI DÜŞÜNMEDE SOYUTLAMAYA DENEYSEL BİR YAKLAŞIM.....	33
4.1 Araştırmanın Kurgusu	34
4.1.1 Yeni başlayanlar için hesaplamalı düşünmede prosedürel soyutlama	34
4.1.2 Geleneksel CAD araçları ile hesaplamalı düşünme	37
4.2 Pilot Çalışmalar	40
4.3 Katılımcılar.....	41
4.3.1 Uygulamaların içeriği ve görevlerin tanımı	43
4.3.1.1 Uygulama 1: SOMA oyun blokları	44
4.3.1.2 Uygulama 2: şekil yapma	46
4.3.1.3 Uygulama 3: yüzeyler	49
4.4 Tasarımcı Bakış Açısıyla Hesaplamalı Düşünmede Soyutlama Becerisinin Değerlendirilmesi	53
4.4.1 Görev tanımı	54
4.4.2 Modül ve komutlarla alıştırma süreci	57
4.4.3 Katılımcılar	58
4.4.4 Veri toplama yöntemi ve kullanılan araçlar	59
4.4.5 Değerlendirme yöntemi.....	59
4.4.5.1 Kodlama şemasının oluşturulması	60
4.4.5.2 Segment değerlerinin tanımlanması	63
4.5 Değerlendirme	67
4.5.1 Görsel hesaplamada kavramsal ve prosedürel soyutlama ilişkisi	67

4.5.2 Tasarım uzmanlığının kavramsal ve prosedürel soyutlamaya etkisi.....	70
4.5.3 Arayüz etkileşimi ve araç kullanımında tasarım uzmanlığının rolü	73
5. SONUÇ VE ÖNERİLER.....	77
KAYNAKLAR.....	81
EKLER.....	87
ÖZGEÇMİŞ.....	97



KISALTMALAR

HD	: Hesaplamalı Düşünme
CT	: Computational Thinking
CAD	: Computer Aided Design
CAM	: Computer Aided Manufacturing
CAE	: Computer Aided Engineering
NAAB	: The National Architectural Accrediting Board
NRC	: National Research Council
HCI	: Human-Computer Interaction
ACARA	: The Australian Curriculum, Assessment and Reporting Authority
CSTA	: Computer Science Teachers Association
ISTE	: International Society for Technology in Education



ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Hesaplamalı düşünmenin bilişsel değerleri (Url-1).....	9
Çizelge 2.2 : Eski HD ve Yeni HD (Denning, 2017; Türkçe'ye tercüme edilmiştir).	12
Çizelge 2.3 : HD kavramları ve karşılıkları (uyarlamadır; Url-4; Automation).....	14
Çizelge 2.4 : Yeni yaklaşımı çerçevesinde yer verilen HD tanımlarında yer alan kavramların dağılımı (2006-2018).....	16
Çizelge 4.1 : Pilot çalışmaların yer aldığı Şekil Yapma Çalıştayı'na ait içerik çizelgesi.	41
Çizelge 4.2 : Deneyde yer alan katılımcılarının demografik dağılımı.....	58
Çizelge 4.3 : Katılımcıların her bir ana hamle için harcamış olduğu ortalama süre (ms).....	67
Çizelge 4.4 : Katılımcıların görev tamamlama süreleri ve gerçekleştirdikleri hamle değerleri	68
Çizelge 4.5 : Hamlelerin içeriğindeki tekrarlara göre tanımlanan segment değerleri ve yüzdelik dağılımı.....	69
Çizelge 4.6 : Katılımcıların kavramsal ve prosedürel soyutlama becerilerinin kıyaslanması.....	70
Çizelge 4.7 : Modül veya şekil türetmek için katılımcıların kullanmış oldukları komutlar ve kullanım sıklıkları.....	70
Çizelge 4.8 : Katılımcıların ana hamleler içinde türettikleri ve kullandıkları original şekil sayısı.....	71
Çizelge 4.9 : Katılımcıların görev süresince komutlarla ilişkilendirerek türettikleri ve kullandıkları ara şekiller ve kullanım sıklıkları.	72
Çizelge 4.10 : Katılımcıların arayüz ile etkileşimi süresince gerçekleştirdikleri eylemler ve görev tamamlama sürelerinin karşılaştırılması.	73
Çizelge A.1 : Pilot çalışmalar sonunda gerçekleştirilen uygulama anket soru ve cevapları.....	89



ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Google - Blockly web uygulaması örneğinden bir ekran alıntısı (Url-3)...	9
Şekil 2.2 : LOGO Programlama'ya giriş eğitim videosundan ekran alıntısı (Url-4).	10
Şekil 2.3 : Hesaplamanın Zihinsel Araçları; Wing'in (2008) tanımından uyarlamadır.	17
Şekil 2.4 : UNC Charlotte Mimarlık Okulu, Bilgisayar ve Hesaplama Programı, (Senske, 2011).	20
Şekil 2.5 : İTÜ Mimarlık Fakültesi, imaride Hesaplamalı Tasarım Araçları ve Yöntemlerine Giriş dersi katalog formu, (Url-8, 2019).	20
Şekil 2.6 : WebofScience veri tabanında hesaplamalı düşünmenin mimarlık/tasarım eğitimi ile ilişkilendirildiği yayınların sayıları ve konu alanları.	21
Şekil 3.1 : Sanatçı soyutlamasına örnek olarak Piet Mondrian'ın çalışmaları (Url- 9).	28
Şekil 3.3 : Görsel düşünürlerin örüntü tanımaya ihtiyaç duyduğu eylemler (Ware, 2008).	29
Şekil 3.4 : Basit geometrik şekillerden oluşan örüntüler.	30
Şekil 4.1 : Bayrak Kodlama egzersizlerinden ekran alıntıları, Google Grasshopper Mobil Uygulaması, 2019.	35
Şekil 4.2 : Scratch uygulamasında bir çizim problemi ve üretilen farklı çözüm önerileri (Url-10).	36
Şekil 4.3 : Rhino modelleme programında, parametrik modelleme eklentisi Grasshopper'dan bir uygulama örneği (Khabazi, 2012).	38
Şekil 4.4 : “İki Lego© parçasını gerçek hayatta ve dijital ortamda birleştirmek, uzamsal ilişkileri farklı detaylarda planlamayı gerektirir.” (Erhan, Youssef ve Berry, 2012; s:384).	39
Şekil 4.5 : AutoCAD ortamında seçilen bir LEGO© parçasını diğeri ile birleştirmede yer alan talimat dizisi.	40
Şekil 4.6 : Bilgisayar laboratuvarında çalıştay uygulama ortamından bir görüntü. ..	42
Şekil 4.7 : Pilot çalışmalarda yer alan katılımcıların öz-yönlendirme ile öğrendikleri bilgisayar programlarının grafik dağılımı.	42
Şekil 4.8 : Pilot çalışmalarda yer alan katılımcıların iki boyutlu uygulamalarda kullandıkları bilgisayar programlarının grafik dağılımı.	43
Şekil 4.9 : SOMA Oyun blokları, (Url-11).	44
Şekil 4.10 : Blok Modelleme ve Kule Oyunu sürecinden ekran alıntısı.	45
Şekil 4.11 : Şekil Yapma görevi için oluşturulan araç paleti ve altlıktan bir ekran alıntısı.	48
Şekil 4.12 : Öğrencilerin çizgi geometrileri farklılaşan algısal yaklaşımlarla ele alış biçimlerine dair iki farklı uygulama örneği.	48
Şekil 4.13 : Yüzeyler görevi için tanımlanan uygulama altlığından bir ekran alıntısı.	50
Şekil 4.14 : Yüzeyler için tanımlanan ikinci görevde sunulan geometrik şekil.	50
Şekil 4.15 : Yüzey Oyunu'nda oyuncuların hamlelerindeki şekil seçimlerine bağlı olarak karmaşılaşan örüntüler oluşturmaya dönüşmesine bir örnek. .	51

Şekil 4.16 : Tanımlanan geometrik şeklin minimum modellenmesi için kullanılacak hamle ve modül birleşimlerini gösteren ideal çözümlerden biri.	56
Şekil 4.17 : Uygulamada katılımcılara sunulan altlık ve uygulama arayüzünden ekran alıntısı.	57
Şekil 4.18 : Tasarımcı olmayan katılımcılardan S6'nın uygulama sürecinden örnek.	57
Şekil 4.19 : Tobii ProLab Kullanıcı arayüzünden bir ekran alıntısı.	59
Şekil 4.20 : Kodlama şeması için tanımlanan değerler.	60
Şekil 4.21 : Modelleme sürecinde katılımcıların eylemlerinin analizinde zaman çizelgesi üzerinde tanımlanan değerlerin grafik açılımı.	62
Şekil 4.22 : S1'in ana hamlelerindeki kodlama içeriğine bağlı olarak tanımlanan 'segment' ve 'çoklu segmentler'den bir örnek.	63
Şekil 4.23 : Uygulama sürecinde katılımcıların gerçekleştirdikleri ana hamleler ve segment değerlerinin karşılaştırması	65
Şekil 4.24 : Katılımcıların uygulama süresince her bir ana hamle için harcadıkları sürenin uygulama sürecindeki yüzdelik dağılımı.	75
Şekil B.1 : Final çalışmaları için katılımcılara uygulanan ankette yer alan ön-değerlendirme soruları ve yanıtları.	91
Şekil B.2 : Final çalışmaları için katılımcılara uygulanan ankette yer alan ön-değerlendirme soruları ve yanıtları.	92
Şekil C.1 : Tasarımcı katılımcıların eylemlerinin içeriğinin grafik ifadesi.	93
Şekil C.2 : Tasarımcı olmayan katılımcıların eylemlerinin içeriğinin grafik ifadesi.	95

TASARIMCI BAKIŞ AÇISIYLA HESAPLAMALI DÜŞÜNMEDE PROSEDÜREL SOYUTLAMANIN DEĞERLENDİRİLMESİ İÇİN BİR YÖNTEM

ÖZET

Dijital araçlar, robotik sistemler ve yapay zekâ uygulamaları gibi sıkça karşılaştığımız teknoloji ürünleri, günlük hayatımıza yerleştikçe bilişsel eylemlerimizi de belirgin biçimde etkilemektedir. Bu süreçte, bilişim teknolojilerinden en üst düzeyde verim elde edebilmek ve bu teknolojilerin gelişimine katkıda bulunabilmek adına, eğitim alanında farklı düşünme biçimleri karşımıza çıkmaktadır. Bu kapsamda en sık karşılaştığımız kavramlardan biri ise şüphesiz, Hesaplamalı Düşünmedir (HD).

Eğitim alanında ilk defa programlama dillerinin pratiği çerçevesinde tanıdığımız bu düşünme biçimi; bugün yeni yaklaşımı çerçevesinde, farklı eylemler ve kavramların nitelikli birlikteliği ile ortaya çıkan bir bilişsel bir zihin aracı olarak nitelendirilmektedir. Bu kapsamda, Bilgisayar Bilimleri dışında, bilişim teknolojilerini aktif olarak kullanan çoğu disiplin tarafından yirmi birinci yüzyılın temel becerilerinden biri olarak kabul görmektedir. Farklı dersler ya da konu alanlarına uyumu söz konusu olduğunda ise HD, genellikle ‘Soyutlama (Abstraction), Örüntü Tanıma (Pattern Recognition), Ayrıştırma (Decomposition) ve Algoritmik Düşünme (Algorithmic Thinking)’ olmak üzere dört bilişsel değeri ile ele alınmaktadır.

Yeni yaklaşımıyla, mimarlık eğitiminde, bilişim teknolojilerinin öğretim kültürü altında, HD’ye yer vermenin hem eğitimci hem de öğrenci açısından stratejik önem taşıdığını söylemek mümkündür. Sunduğu dört temel bilişsel değeri ile bu düşünme biçimi, mimarlık öğrencilerine tasarım, temsil ve üretim süreçlerinde doğru araç ve yöntemleri seçmelerine ve öğrencilerin karşılaştıkları problemleri ya da durumları formüle ederek ve hesaplama ortamına uyarlamaları için onlara eğitilmiş bir göz kazandırmaya yardımcıdır. Bu doğrultuda, söz konusu bilişsel değerlerin tasarımcı bakış açısıyla ve görsel düşünörlere yönelik öğrenim deneyimleriyle teşvik edilmesi, mimarlık eğitiminde HD’nin pedagojik uyumu çerçevesinde dikkate alınmalıdır ve bu kapsamda soyutlamanın, HD’nin diğer bileşenleri arasından öne çıktığını belirtmek mümkündür.

‘Soyutlama’ veya ‘soyut düşünme’ hem HD sürecinde ihtiyaç duyulan hem de tasarım eğitiminde teşvik edilen temel becerilerdendir. Ancak, tasarım bilimlerinin görsel düşünörlürleri için sunduğu soyutlama yaklaşımı, HD bağlamında ihtiyaç duyulan soyutlama yaklaşımına kıyasla farklılık ihtiva eder. HD’de soyutlamaya, yöntemler ve işlemler üzerinden genellemeler yaparak prosedürel düzeyde ihtiyaç söz konusudur. Tasarım eğitiminde ise soyutlama becerisi, daha çok algısal yaklaşımlar üzerinden yapılan genellemelere ve somutlaşmış (embodied) öğrenim deneyimleriyle, kavramsal düzeyde teşvik edilmektedir.

Dolayısıyla, tasarımda bilişim araçlarıyla çalışan bir mimarlık öğrencisinin hem kavramsal hem de prosedürel düzeyde soyutlama becerisine sahip olması oldukça önemlidir. Bu durum aynı zamanda, bir tasarımcının hesaplamalı düşünme sürecinde kullandığı soyutlama becerisinin, kavramsal ve prosedürel olmak üzere, farklı düzeylerde değerlendirilmesi ve geliştirilmesi gerektiğinin göstergesidir. Mimarlık eğitiminde HD'nin öğretim kültürü çerçevesinde, özelleştirilebilir/bireyselleştirilebilir öğrenim deneyimleri sunabilmenin pedagojik açıdan önemini ortaya koymaktadır.

Bu tezde de mimarlık eğitiminde HD'nin öğretim kültürü ve pedagojik uyumu, soyutlama, örüntü tanıma, ayırıştırma ve algoritmik düşünme çerçevesinde bilişsel yönüyle ele alınmıştır. Bu kapsamda, tasarımcı bakış açısıyla soyutlamanın görsel hesaplama üzerindeki etkisi araştırılmış; mimarlık öğrencilerinin soyutlama becerilerini kavramsal ve prosedürel yönden ölçmek ve değerlendirmek üzere bir yöntem geliştirilmiştir. Geliştirilen bu yöntem, farklı disiplinlerden katılımcıların yer verildiği bir deney uygulamasında kullanılmıştır. Toplamda altı katılımcı ile gerçekleştirilmiş olan bu deneyde yer verilen uygulamanın içeriğinin ve değerlendirme yönteminin geliştirilmesinde, yeni başlayanlar için HD'ye yönelik uygulama örnekleri ve mimarlık öğrencileriyle gerçekleştirilen bir dizi pilot çalışma referans alınmıştır.

Yöntemin geliştirilmesi için 'Yeni Başlayanlar İçin HD' kapsamında hazırlanan eğitim uygulamalarında soyutlama becerisinin değerlendirilmesine yönelik örnekler incelenmiş; bu örneklerde kullanılan değerlendirme yöntemlerinin, Bilgisayar Destekli Tasarım (CAD) ortamında uygulanabilmesi için bir dizi pilot çalışma gerçekleştirilmiştir.

Geliştirilen bu yöntem, farklı tasarım uzmanlığına sahip (tasarımcı ve tasarımcı olmayan) katılımcılar ile gerçekleştirilen uygulamalı bir deneyde kullanılmıştır. Deneyde, katılımcılara iyi-tanımlanmış bir problem niteliğinde, geleneksel CAD programında hazırlanmış bir şekil yapma görevi verilmiştir. Bu görev, farklı tasarım uzmanlığına sahip görsel düşünürleri kıyaslamaya engel teşkil etmeyecek şekilde (katılımcıların verilen şekil ve komutları tanıma, arayüz ve araç kullanma ve uzamsal becerilerde yetkinlik faktörlerinin dikkate alınmasıyla) kurgulanmıştır.

Her bir katılımcının modelleme süreci, araç ve arayüz kullanımları çerçevesinde analiz edilmiştir. Veri olarak katılımcıların uygulamaları sırasında alınan video ekran ve ses kayıtları, ekran üzerindeki göz hareketleri ve araç kullanım kayıtlarından yararlanılmıştır. Katılımcıların göz hareketleri ve arayüz etkileşim kaydı için Tobii Pro X2-30 ekrana bağlı göz takip cihazı, analizi için ise Tobii ProLab 30 günlük deneme sürümü kullanılmıştır. Kullanılan bu araçlar, katılımcıların modelleme sürecinde türetmiş oldukları şekilleri, kullanmış oldukları komutları ve araçları saptamaya yardımcı olmuştur.

Değerlendirmede, araştırmanın bilişsel ve sistematik analizinden elde edilen bulgulardan yararlanılmıştır. Bu bağlamda, katılımcıların verilen uygulamaları tamamlama süreleri ve araç kullanımlarının yanında; arayüz etkileşimleri sırasındaki gerçekleştirmiş oldukları geometrik şekil seçimleri ve komut kullanımları ile ortaya çıkarmış oldukları soyutlama örüntüleri karşılaştırılmıştır. Katılımcıların soyutlama örüntüleri, uygulama sırasında kullanmış oldukları (türetmek için seçtikleri şekil, türetmek için kullandıkları komut ve türettikleri şekil olmak üzere) komutlar ve şekiller üzerinden oluşturulan bir kodlama şeması ve bu şema üzerinden tanımlanan segment değerleri ile ortaya çıkarılmıştır. Bu doğrultuda, katılımcıların prosedürel soyutlama becerileri, segment değerlerine bağlı olarak nicel yönden karşılaştırılmıştır.

AN EVALUATION METHOD FOR A DESIGNER'S PROCEDURAL ABSTRACTION IN COMPUTATIONAL THINKING

SUMMARY

We are currently facing a paradigm shift towards a computing and coding culture. From digital tools to artificial intelligence, computational technologies are visibly affecting our everyday life and shaping our cognitive development. With this paradigm shift, we are imposed on different modes of thinking; and without a doubt, Computational Thinking is the highest in-demand. For a better understanding and use of computational technologies in education, Computational Thinking (CT) has been pursued as a fundamental skill among disciplines over a decade.

Despite its former use in programming practices, CT is currently used as a mind tool, or, as a mental activity for formulating a problem to admit a computational solution. With its new approach, CT is addressed as a collection of mental tools and concepts that are borrowed from computer sciences. When it comes to integration of this concept into different disciplines, CT is usually introduced with its four cognitive values: 'Abstraction, Pattern Recognition, Decomposition, and Algorithmic Thinking.'

On the other hand, CT is not about developing some knowledge of these aspects. As a soft skill, it requires putting the knowledge into action, using, transferring, and adapting it into different problems or situations. Hence, outside computer sciences, developing a better understanding of its cognitive aspects and their counterparts is essential. Within this regard, architecture is one of the disciplines that requires careful consideration of CT's integration into its education curricula.

In architectural education, CT partakes a significant role in the culture of teaching and learning computational technologies. Within the widening use of customized design tools and programs in architectural design and production, architectural students are likely to become digital literates. And the workflow behind these technologies involves different sets of computational operations, which requires designers to think computationally. Hence, with its four cognitive aspects, CT provides an educated eye to architecture students; it helps them formulate their problems or situations and adapt themselves to work efficiently with computational design technologies. However, when it comes to the convergence of CT's cognitive aspects to architectural education, little attention has been paid to a designer's approach to the concept of abstraction in computational thinking.

The conception of 'abstraction' or 'abstract thinking' is one of the fundamentals skills in design sciences to be acquired in their early years of education. With their unique tools and methods, architecture students are endorsed to practice their abstraction skills at conceptual level through the years. Therefore, as visual thinkers, architectural students are expected to develop good abstraction skills. However, there is a subtle difference when it comes to use of abstraction skill towards computational thinking.

For designers, the use of computational design technologies acquires the use of abstraction at different levels in different forms of representation. In most cases, the workflow behind computer-aided design technologies requires making abstractions at different levels and with different forms of abstraction. Especially, when programming, avoiding repeating code, making abstractions and generalizations at a procedural level, makes projects easier to program and maintain. Hence, having considered the designer as a visual thinker, this study sheds light on a designerly use of abstraction in CT. It offers an assessment method for the designers' use of procedural abstraction towards visual computing.

Since this is a novel analysis for the designers' use of abstraction skills towards a CAD modeling framework, a vital first step is to establish a coding scheme that captures behaviors at a broad enough level to be applicable to related research and specific enough to identify behaviors that are unique to visual thinkers. In this regard, the assessment method was developed through a series of pilot studies with architecture students by specifying the designers' actions in the CAD modeling process; then applied in the experiment with the participation of six subjects.

As an experiment, all subjects were given a 2D CAD modeling task with the limitation of certain CAD tools and commands. The subjects were expected to think about what kinds of computational operations would be useful or necessary for the modeling of the given geometry with the given modules.

For a systematic analysis of the subjects' modeling processes, the video-screen recordings, command history scripts, and interface interactions of the subjects with the computer hardware were collected to be used as data. Following the completion of all the experiments, the collected data was encoded using Tobii ProLab 30-day Trial Software.

By looking at the experiments, it can be said that working with computational technologies requires making abstractions at a different level. And, the use of assessment method has revealed that making abstractions at a procedural level has a significant impact on the designer's visual computing. Regarding the subjects' performances in the CAD modeling tasks, it can be said that the better use of CT relies heavily on finding a balance between different levels of abstraction, or in other words, making good generalizations. For a visual thinker, working with computational technologies should not be seen as a matter of collecting or manipulating a random or an infinite use of data. In many ways, it should be matter of recognizing visual and behavioral patterns.

The content of the thesis is given as follows.

Chapter 1 includes the aim and scope of the thesis towards a problem definition. It briefly discusses the new approach to CT and its integration into architecture education by addressing the challenges in the current culture of teaching design technologies.

Chapter 2 reviews the existing literature of computational thinking in education and scrutinizes the teaching culture of CT for beginners outside Computer Sciences. By addressing the differences between the traditional and the new approaches to the concept of CT, it discusses the teaching and learning culture of computational technologies in architectural education under some of the exemplary work. It evaluates the pedagogical alignment of CT as a cognitive tool in architectural education.

Chapter 3 presents a curation of existing knowledge of CT concepts (abstraction, pattern recognition, decomposition, and algorithmic thinking) and compare their conceptual counterparts towards the computer and design sciences.

Chapter 4 includes three subsections: the development of the assessment method and content of the task experiments, the analyses of the experiments, and the findings and results. In the first section, the development of the task experiments was presented through a series of pilot studies. And the development of the assessment method for the visual thinker's process of computational thinking was presented towards several case studies from the Human-Computer Interaction field. In the second section, the analysis of the two-stage task experiment was exhibited. In the third section, the results and findings of the task experiments were discussed.

Chapter 5 concludes the results of the experiments within the research scope. Additionally, it includes several suggestions for the assessment method's development into a digital education tool for visual thinkers.





1. GİRİŞ

1.1 Problemin Tanımı ve Tezin Amacı

Dijital araçlar, robotik sistemler ve yapay zekâ uygulamaları gibi sıkça karşılaştığımız teknoloji ürünleri, günlük hayatımıza yerleştikçe bilişsel eylemlerimizi de belirgin biçimde etkilemekte; bizleri, hesaplama ve kodlama kültürüne doğru bir paradigmaya yönlendirmektedir.). Bu paradigma, bir yandan kullandığımız teknolojinin bilişsel sürecimizi desteklemek üzere ‘beyin dostu ’olmasını (Dror, 2011; s:2); diğer yandan zihinsel eylemlerimizin bilişim ortamına uyum sağlayacak şekilde gelişim göstermesini gerektirmektedir. (Weisberg & Reeves, 2013; s:17). Dolayısıyla bizler, kullandığımız bilişim araçlarıyla verimli çalışabilmek ve bu araçların gelişimine katkıda bulunabilmek adına farklı düşünme biçimlerini ödünç almaktayız ve bu kapsamda eğitimde en fazla referans gösterdiğimiz düşünme biçimlerinden biri, hiç şüphesiz ki Hesaplamalı Düşünmedir (HD).

Eğitim alanında ilk defa programlama dillerinin pratiği çerçevesinde tanıdığımız bu düşünme biçimi; bugün yeni yaklaşımı çerçevesinde, farklı eylemler ve kavramların nitelikli birlikteliği ile ortaya çıkan bir bilişsel bir zihin aracı olarak nitelendirilmektedir. Bu kapsamda, Bilgisayar Bilimleri dışında, bilişim teknolojilerini aktif olarak kullanan çoğu disiplin tarafından yirmi birinci yüzyılın temel becerilerinden biri olarak kabul görmektedir. Farklı dersler ya da konu alanlarına uyumu söz konusu olduğunda ise HD, genellikle ‘Soyutlama (Abstraction), Örüntü Tanıma (Pattern Recognition), Ayrıştırma (Decomposition) ve Algoritmik Düşünme (Algorithmic Thinking) ’olmak üzere dört bilişsel değeri ile ele alınmaktadır. Bu doğrultuda mimarlık eğitiminde, bilişim teknolojilerinin öğretim kültürü altında, HD’ye yer vermenin hem eğitimci hem de öğrenci açısından stratejik önem taşıdığını söylemek mümkündür.

Mimarlık öğrencilerinin, eğitimleri süresince bilişim teknolojilerinin kullanımına doğru araçlar ve yöntemlerle yer vermeleri; bu doğrultuda eğitimlerinin ilk yılından itibaren deneyim ve beceri kazanmaları, pedagojik açıdan oldukça önemlidir. Ancak, ilk örneklerine kıyasla, tasarımda bilişim teknolojileri oldukça fazla çeşitlilik gösterir ve bu çeşitlilik içinde tasarımcı adayının, karşılaştığı problemleri ya da durumları formüle edebilmesi ve bilişim ortamına uyarlayabilmesi için, öncelikle ‘eğitimli bir göz’e sahip olması gerekir. Daniel Reisberg’in ifadesiyle (2015) ‘eğitimli bir göz ’(an educated eye), “bilgi ve deneyimi, başkalarının göz ardı ettiği şeyleri görmede, başkalarına görünmeyen örüntüleri büyük ölçüde tespit etmede, bazı durumlarda ise yapılabilecek yeni kombinasyonları seçmede yardımcıdır” (ss:99). Nitekim HD de ihtiva ettiği “Soyutlama, Örüntü Tanıma, Ayırıştırma, ve Algoritmik Düşünme” değerleri ile, mimarlık öğrencilerinin temsil, tasarım ve üretim süreçlerinde, bilişim araçlarıyla verimli ve etkin biçimde çalışmak için ihtiyaç duydukları ‘ eğitimli gözü ’ kazandırabilecek bir zihin aracıdır. Ve bu doğrultuda, söz konusu bilişsel değerlerin tasarımcı bakış açısıyla ve görsel düşünürlere yönelik öğrenim deneyimleriyle teşvik edilmesi, mimarlık eğitiminde HD’nin pedagojik uyumu çerçevesinde dikkate alınmalıdır; ve bu kapsamda soyutlamanın, HD’nin diğer bileşenleri arasından öne çıktığını belirtmek mümkündür.

‘Soyutlama’ veya ‘soyut düşünme’ hem HD sürecinde ihtiyaç duyulan hem de tasarım eğitiminde teşvik edilen temel becerilerdendir. Ancak, tasarım bilimlerinin görsel düşünürlere için sunduğu soyutlama yaklaşımı, HD bağlamında ihtiyaç duyulan soyutlama yaklaşımına kıyasla farklılık ihtiva eder. HD’de soyutlamaya, yöntemler ve işlemler üzerinden genellemeler yaparak prosedürel düzeyde ihtiyaç söz konusudur. Tasarım eğitiminde ise soyutlama becerisi, daha çok algısal yaklaşımlar üzerinden yapılan genellemelere ve somutlaşmış (embodied) öğrenim deneyimleriyle, kavramsal düzeyde teşvik edilmektedir.

Dolayısıyla, tasarımda bilişim araçlarıyla çalışan bir mimarlık öğrencisinin hem kavramsal hem de prosedürel düzeyde soyutlama becerisine sahip olması oldukça önemlidir. Bu durum aynı zamanda, bir tasarımcının hesaplamalı düşünme sürecinde kullandığı soyutlama becerisinin, kavramsal ve prosedürel olmak üzere, farklı düzeylerde değerlendirilmesi ve geliştirilmesi gerektiğinin göstergesidir. Mimarlık eğitiminde HD’nin öğretim kültürü çerçevesinde, özelleştirilebilir/bireyselleştirilebilir öğrenim deneyimleri sunabilmenin pedagojik açıdan önemini ortaya koymaktadır.

Bu tezde de mimarlık eğitiminde HD'nin öğretim kültürü ve pedagojik uyumu, soyutlama, örüntü tanıma, ayrıştırma ve algoritmik düşünme çerçevesinde bilişsel yönüyle ele alınmıştır. Bu kapsamda, tasarımcı bakış açısıyla soyutlamanın görsel hesaplama üzerindeki etkisi araştırılmış; mimarlık öğrencilerinin soyutlama becerilerini kavramsal ve prosedürel yönden ölçmek ve değerlendirmek üzere bir yöntem geliştirilmiştir. Geliştirilen bu yöntem, farklı disiplinlerden katılımcıların yer verildiği bir deney uygulamasında kullanılmıştır.

1.2 Araştırma Yöntemi ve İçerik

Literatüre baktığımızda, mimarlık eğitiminde, HD ve temel bilişsel değerlerine yer veren araştırmaların, ağırlıklı olarak biçim (form) odaklı ve hasta tanımlı (ill-defined) tasarım problemleri üzerinden yürütülen çalışmalar çerçevesinde kurgulandığını söylemek mümkündür. Çoğu araştırmada HD, parametrik tasarım veya evrimsel tasarım gibi özelleşmiş konu başlıkları altında, temel bilişsel değerlerinin nitelikli birlikteliğinden uzakta veya hesaplamalı tasarım programlarının kullanımı çerçevesinde ele alınmaktadır. Ancak mevcut araştırmaların aksine, bu araştırmada HD soyutlama, örüntü tanıma, ayrıştırma ve algoritmik düşünme eylemlerinin birlikteliğini gerektiren bir zihin aracı olarak nitelendirilmiştir. Bu doğrultuda, farklı tasarım uzmanlığına sahip katılımcıların görsel hesaplama süreçlerinin bilişsel ve sistematik analizine ihtiyaç duyulmuş; bunun için, söz konusu dört bileşenin birlikte kullanımını gerektiren bir görsel hesaplama problemi karşısında, katılımcıların bilişsel süreçlerini analiz etmek üzere bir deney gerçekleştirilmiştir.

Farklı disiplinlerden toplamda 6 katılımcı ile gerçekleştirilmiş olan bu deneyde yer verilen uygulamanın içeriğinin ve değerlendirme yönteminin geliştirilmesinde, yeni başlayanlar için HD'ye yönelik uygulama örnekleri ve mimarlık öğrencileriyle gerçekleştirilen bir dizi pilot çalışma referans alınmıştır.

Gerçekleştirilen deney için katılımcıların belirlenmesinde, tasarım uzmanlıkları, bilgisayar kullanım becerileri ve uzamsal becerileri dikkate alınmıştır. Bu bağlamda, öz-yönlendirmeli öğrenme (self-regulated learning) deneyimleriyle bilişim teknolojilerini kullanımı yetkinlik kazanmış olan üç mimarlık öğrencisinin performansı, tasarım disiplinleri dışından lisans düzeyinde üç katılımcı ile karşılaştırılmıştır.

Uygulama olarak katılımcılara, Bilgisayar Destekli Tasarım (CAD) ortamında iki boyutlu katı nesne modelleme (solid object modeling) görevi tanımlanmış; her bir katılımcının modelleme süreci, araç ve arayüz kullanımları çerçevesinde analiz edilmiştir. Veri olarak katılımcıların uygulamaları sırasında alınan video ekran ve ses kayıtları, ekran üzerindeki göz hareketleri ve araç kullanım kayıtlarından yararlanılmıştır. Katılımcıların göz hareketleri ve arayüz etkileşim kaydı için Tobii Pro X2-30 ekrana bağlı göz takip cihazı, analizi için ise Tobii ProLab 30 günlük deneme sürümü kullanılmıştır. Kullanılan bu araçlar, katılımcıların modelleme sürecinde türetmiş oldukları şekilleri, kullanmış oldukları komutları ve araçları saptamaya yardımcı olmuştur.

Değerlendirmede, araştırmanın bilişsel ve sistematik analizinden elde edilen bulgulardan yararlanılmıştır. Bu bağlamda, katılımcıların verilen uygulamaları tamamlama süreleri ve araç kullanımlarının yanında; arayüz etkileşimleri sırasındaki gerçekleştirmiş oldukları geometrik şekil seçimleri ve komut kullanımları ile ortaya çıkarmış oldukları soyutlama örüntüleri karşılaştırılmıştır.

Katılımcıların soyutlama örüntüleri, uygulama sırasında kullanmış oldukları (türetmek için seçtikleri şekil, türetmek için kullandıkları komut ve türettikleri şekil olmak üzere) komutlar ve şekiller üzerinden oluşturulan bir kodlama şeması ve bu şema üzerinden tanımlanan segment değerleri ile ortaya çıkarılmıştır. Bu doğrultuda, katılımcıların prosedürel soyutlama becerileri, segment değerlerine bağlı olarak nicel yönden karşılaştırılmıştır.

1.3 Araştırmanın Özgün Değeri ve Literatüre Katkısı

Son on yıl içinde hesaplamalı düşünmenin kazanmış olduğu yeni yaklaşım ve öğretim kültüründeki radikal değişiklikler, zorunlu eğitimde olduğu kadar bilgisayar bilimleri dışındaki disiplinler için de büyük yankı uyandırmıştır. Mimarlık dahil olmak üzere pek çok disiplin için bu durum, HD ve temel bilişsel değerlerinin müfredata yansımaları biçimi ve pedagojik uyumu üzerine araştırmaları yeniden gündeme getirmiştir.

Ancak, mimarlık eğitimi çerçevesinde ne HD'nin ne de bilişsel değerlerinin eğitimciler ve öğrenciler için yabancı kavramlar olduğunu söylemek mümkündür. Yalnızca, mevcut araştırmaların HD'nin öğretim kültürüne ağırlıklı olarak geleneksel yaklaşımı çerçevesinde yer vermiş olmaları, bu kavramı mevcut mimarlık eğitim müfredatında bir zihin aracı olarak nitelendirmeyi ve temel bir beceri olarak teşvik etmeyi oldukça zorlaştırmıştır. Dolayısıyla, HD'yi temel bilişsel değerleri ile birlikte, eğitim müfredatına daha pragmatist bir çerçevede kazandırmayı amaçlayan bu araştırmanın, sunduğu kavramsal çerçeve ve değerlendirme yöntemi ile söz konusu pedagojik boşluğu doldurmaya önemli katkıda bulunması beklenmektedir.

Tasarımcının prosedürel soyutlama becerisini nicel yönden değerlendiren bu araştırma, sunduğu uygulama ve değerlendirme yöntemi ile literatürdeki özgün araştırmalardan biridir. Bu yönüyle araştırmanın, mimarlık eğitiminde bilişim araçlarının öğretim kültürü çerçevesinde, bireye farklı düzeylerde soyutlama becerisi kazandıran ve bunu denetleyen bir öğrenim mekanizmasının geliştirilmesi için nitelikli bir kaynak oluşturduğunu söylemek mümkündür. Ayrıca, görsel düşünürün HD sürecine bilişsel yönüyle yer vermiş olması nedeniyle, yapılan araştırmanın hem daha iyi öğrenim deneyimleri tasarlamaya yardımcı olacağı hem de HD'nin mimarlık eğitimine pedagojik uyumu konusunda çok daha somut bir çerçeve çizeceği öngörülmektedir.



2. HESAPLAMALI DÜŞÜNMENİN MİMARLIK EĞİTİMİNDEKİ ÖĞRETİM KÜLTÜRÜ VE PEDAGOJİK UYUMU

2.1 Yirmi Birinci Yüzyıl Becerisi Olarak Hesaplamalı Düşünme

"Bir bilgisayar olarak zihin" fikrinden yola çıkan Herbert Simon ve Alan Newell, sundukları "genel problem-çözme modeli" (1971) ile hesaplayarak düşünme eylemini işlevsel seviyede bilişsel bilim literatürüne kazandıran isimlerdir (Weisberg & Reeves, 2013; s:17). Fakat eğitimdeki ilk uygulama örneklerine bakıldığında, bu düşünme biçiminin, ağırlıklı olarak sayısallaştırma yoluyla problem çözmede kullanıldığını ve bir tür mesleki teknik beceri (hard skill) olarak nitelendirildiğini söylemek mümkündür. Çağdaş eğitim sistemindeki Hesaplamalı Düşünme, çok daha geniş yaş aralığı ve kullanıcı grubunu hedef alan, disiplinler arası geçerlik kazanmış bir zihin aracı olarak karşımıza çıkar. Bu kapsamda özellikle bilişim teknolojilerindeki gelişmeler, statik hesaplama esnek hesaplama modellerine doğru geçiş, insan-bilgisayar etkileşimindeki değişim, HD'ye olan geleneksel tutumun değişmesinde etkili olmuştur.

HD'ye yeni yaklaşımını kazandıran en önemli isim ise şüphesiz Jeannette Wing olmuştur. The Association of Computing Machinery'de yer alan "Computational Thinking" (2006) makalesi ile Wing, bu düşünme biçimine "bilgisayar biliminin temel kavramlarını kullanarak problemleri çözmek, sistemleri tasarlamak ve insan davranışını anlamak" olarak yeniden anlam kazandırmıştır (s:33). Wing'e göre (2017; s:8) HD, "bir problemi formüle ederken, insan ve makine zekâsını (the intelligence of human and machines) bir araya getirerek hesaplamalı bir çözüm üretmeye yarayan bir zihin faaliyetidir." Yeni yaklaşımı ile bu kavram, makineye özgü bir hesaplama becerisi değil; sıralı ve soyut düşünmeyi, farklı eylem ve becerileri birlikte kullanmayı gerektiren bilişsel bir zihin aracıdır. Bu yönüyle, insan zihni ve teknolojiyi sistematik bir çerçevede ortak bir paydada buluşturma konusunda potansiyel sahibidir. Nitekim, Bilişsel Bilimde bu tür zihin araçları, öğrenmeye yardımcı olmanın yanında (Jonassen, 1992), bireyin düşünme sürecini destekleme, yönlendirme ve genişletmede rol oynarlar (Derry, 1990).

Bu bağlamda HD, bilişsel gelişimimize destekte bulunan bir yirmi birinci yüzyıl becerisi olarak zorunlu eğitim sisteminde yer almaktadır. Bilişim ortam ve araçlarının etkin ve verimli kullanımı için temel bir beceri (Khan ve diğ., 2017); bu ortam ve araçların gelişimine katkıda bulunmak üzere derin öğrenme deneyimleri oluşturmaya yardımcı bir düşünme biçimi olarak kabul görmektedir (Czerkowski & Lyman, 2015).

Peter J. Denning’e göre (2017) HD’nin yeni yaklaşımı, geleneksel yaklaşımından oldukça farklıdır: “Geleneksel yaklaşımda programlama becerisinin hesaplamalı düşünmeyi ortaya çıkardığı öngörülürken, yeni yaklaşımda HD’nin temel oluşturduğu belirli kavramlar programlama becerisini kazandırmaktadır; kısaca yeni yaklaşımda, nedenselliğin yönü tersi biçimde değişmiştir (s:37).”

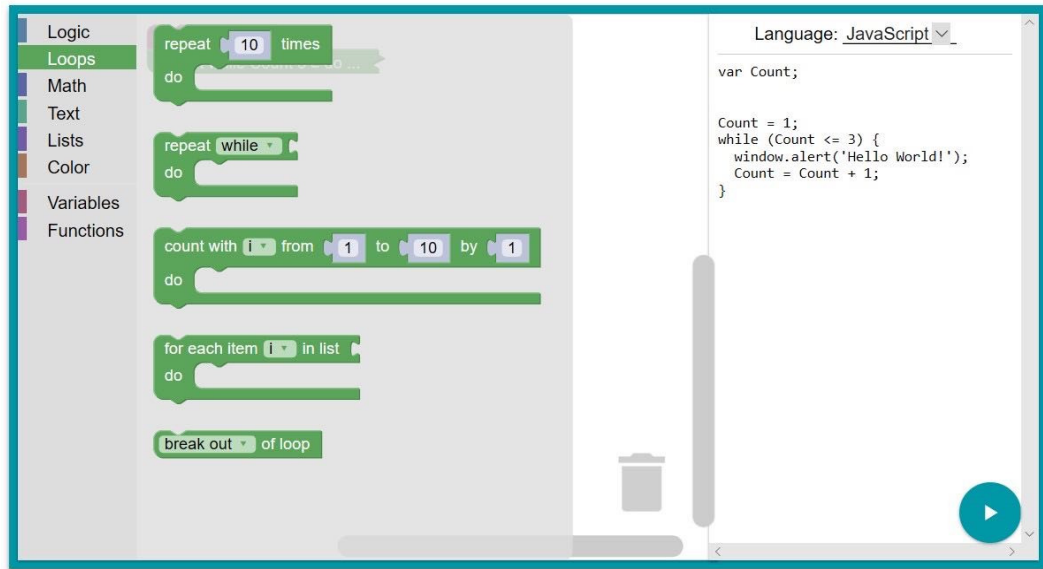
Yeni yaklaşımı ile HD’yi zorunlu eğitim sistemine yerleştirmeyi amaçlayan, kapsamlı çalışmalardan birine Birleşik Devletler Ulusal Araştırma Konseyi’nce (NRC, 2008) gerçekleştirilen “On the Scope and Nature of Computational Thinking” ve “On the Pedagogical Aspects of Computational Thinking” isimleriyle gerçekleştirdiği çalıştay dizisini örnek göstermek mümkündür Aralarında Wing’in de yer aldığı, bilgisayar ve eğitim bilimlerinden önemli isimlerden oluşan bir heyetin yürütücülüğünde gerçekleştirmiş bu çalıştaylarda, HD’nin bilişsel gelişime katkısı, hangi temel kavramlar çerçevesinde şekillendiği, ilköğretim eğitim müfredatı içinde ne tür öğrenim deneyimleriyle sunulması gerektiği masaya yatırılmıştır (NRC, 2010). Sunulan ilk çalıştay raporuna göre heyet, HD’nin birtakım eylem ve becerilerin birlikteliğini içeren bilişsel bir süreç olduğu konusunda hemfikiridir (NRC, 2010; ss:10-11). Diğer yandan aynı raporda, söz konusu eylem ve becerilerin, yalnızca bilgisayar bilimcilerine özgü değil; aynı zamanda farklı disiplin ve yaş gruplarına yönelik, gündelik hayat deneyimleriyle de kazanılabilir ve geliştirilebilir temel bilişsel değerler olmaları oldukça dikkat çekicidir. Wing’in ifadesiyle HD bir zihin aracıdır (mental tool) ve metal araçlardan (metal tool) farklıdır: metal araçlar “problem çözmeye yarayan yazılım ya da donanım uygulamaları iken; zihin araçları “problemi anlamaya ve daha etkili biçimde çözmeye yarayan bilişsel ve düşünsel becerilerdir” (NRC, 2010; s: 10).

Bu kapsamda, soyutlama (abstraction), algoritmik düşünme (algorithmic thinking), örüntü tanıma (pattern recognition) ve ayrıştırma (decomposition) olmak üzere, genellikle dört temel bilişsel değeri ile tanıştırılmaktadır (Url-1; Url-2).

Çizelge 2.1 : Hesaplamalı düşünmenin bilişsel değerleri (Url-1).

Kavram	Tanımı	Örnek
Ayrıştırma (Decomposition)	Problemi küçük aşamalara ayırarak parçalamaktır.	Kahve yapmak için temel adımları düşünmek
Örüntü Tanıma (Pattern Recognition)	Problemi çözmek için kullanılabilecek örüntüleri tanımak ve işlev kazandırmaktır.	Akşam 4'ten sonra içilen kahvenin uyumayı zorlaştırdığını fark etmek
Soyutlama (Abstraction)	Problemi çözmek için ihtiyaç duyulan örüntüleri genellemek veya öncelik kazandırmaktır.	Gündüz katıldığın toplantı yüzünden değil, aldığın kafein yüzünden uyuyamadığını fark etmek
Algoritmik Düşünme (Algorithmic Thinking)	Problemi çözmek veya görmezden gelmek üzere oluşturulan adım adım talimatlar dizisi oluşturmaktır	Bir fincan güzel kahve yapabilmek için gereken aşamaları yazmak

Nitekim 'Eğitim için Google 'veya 'Microsoft Eğitim 'gibi teknoloji ve eğitim sektöründe öncü kuruluşlar, geliştirdikleri stratejiler ile HD'yi teşvik etmekte; Sundukları çevrimiçi uygulamalar ve bu uygulamalara göre hazırladıkları müfredat içerikleriyle, hesaplamalı düşünme ve programlama kültürünü farklı disiplinler ve yaş gruplarına tanıtmakta; bireyin bilişsel gelişimini hedef aldıkları uygulamalarla, HD'nin öğretim kültürünü yeniden biçimlendirmektedirler (Şekil 2.1).

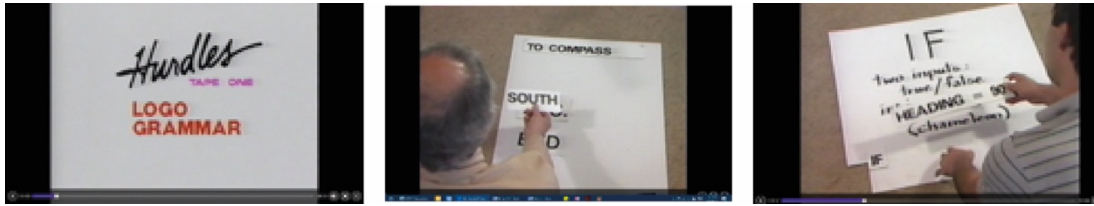


Şekil 2.1 : Google - Blockly web uygulaması örneğinden bir ekran alıntısı (Url-3).

Bu uygulamalarla, bireye kodlama becerisini görsel programlama ile kazandırmak, hesaplamalı düşünmeyi mantık yürütme, ayrıştırma, problem çözme gibi eylemlerle aşlamak hedeflenmekte ve HD bu bilişsel değerlerle esnek bir beceri teşvik edilmektedir.

2.2 Yeni Başlayanlar için Hesaplamalı Düşünmenin Öğretim Kültürü

Bilgisayar Bilimlerinin kazandırmış olduğu Hesaplamalı Düşünme becerisinin farklı disiplinlerin eğitim sistemine yansımaları oldukça yenidir. Bu bağlamda, karşılaştığımız ilk uygulama örnekleri, Alan Perlis'in 'Programlamaya Giriş' (introductory programming) ve Seymour Papert'in 'prosedürel düşünme' (procedural thinking) modelleri olmuştur. Perlis (1962) bu kavramı "problem çözme aracılığı ile bilgisayarların nasıl çalıştığını öğretmek" için kullanmışken; Papert (1981) "ilköğretim öğrencilerine LOGO programlama dili ile prosedürel düşünceyi tanıtmak" amacıyla kullanmıştır. Ancak dönemin teknoloji imkânları ve HD'nin bilişsel araştırmalardaki niceliği göz önüne alındığında, her iki örnek için de HD'ye olan pedagojik yaklaşımın oldukça geleneksel çerçevede kaldığını söylemek mümkündür. Her ne kadar Papert'in çalışmalarında, analog uygulamalarla ilköğretim seviyesini hedef almış olması, HD'nin farklı disiplinler ve eğitim seviyeleri içinde teşvik edilebilir bir beceri olduğunu gösterse de bu uygulamaların oldukça kısıtlayıcı ve geleneksel bir tutum arz ettiği söylenebilir (Şekil 2.2).



Şekil 2.2 : LOGO Programlama'ya giriş eğitim videosundan ekran alıntısı (Url-4).

Nitekim Voogt ve diğ. (2015) bu tür örneklerde, HD'nin bilişsel katkısına yönelik çelişkili çıkarımlar gösterdiğini ifade etmiştir. Bir yandan Kurland ve Pea (1985) ilköğretim öğrencilerine uyguladıkları benzer çalışmalarda, LOGO programlamanın 'planlama becerileri' yönünden, belirgin bir katkısının olmadığını gözlemlemişken; diğer yandan Klahr ve Carver (1988) LOGO programlama sayesinde, aynı seviyedeki öğrencilerin 'hata ayıklama' (debugging) konusunda gelişim gösterdiğini raporlamıştır (Voogt ve diğ., 2015).

Öte yandan, zamanla teknik hesaplama esnek hesaplama modellerine olan yönelim, insan-bilgisayar etkileşimi (Human-Computer Interaction) araştırmalarındaki disiplinlerarası çalışmaların yaygınlaşması; eğitim alanında da ‘hesaplama=programlama’ anlayışını geride bırakmaya yardımcı olmuştur. Bu durum HD’yi, esnek beceri kazandırmaya yardımcı bilişsel bir değer olarak gündeme getirmiştir. Özellikle bilişim teknolojilerindeki gelişimi takiben, farklı programlama dillerinin görsel programlama araçları ile desteklenmesi, HD’nin bilgisayar bilimleri dışındaki disiplinlerde de (öğrenilebilir ve öğretilebilir) teşvik edilebilir hale gelmesine yardımcı olmuştur.

Bu çerçevede, HD’yi “bilgisayar bilimlerindeki temel kavramlardan yararlanarak problem çözmek, sistemleri tasarlamak ve insan davranışını anlamak” olarak yeniden tanıştırmış olan Wing (2006, s:33); bu yaklaşımıyla, HD’yi eğitimde disiplinler arası seviyeye ulaştıran ilk ve en etkili isim olmuştur (Özçınar, 2017). Yeni yaklaşımı ile HD, “problemler ve çözümleri formüle etmede kullanılan bir düşünme sürecidir ve bu süreç, tanımlanan problem veya çözümlerin bir bilgi-işlemci (insan veya makine) tarafından etkin şekilde işlenmesine yardımcıdır” (Wing, 2017). Dolayısıyla, çağdaş eğitim sistemi, hesaplamalı sistemler ve araç kullanımına yönelmek yerine, hesaplamanın ilkeleri ve yöntemlerine, bilimsel ve kültürel değerleri ile ilgilenmeye ihtiyaç duymaktadır (Nardelli, 2019). Çünkü bir zihin aracı olarak bu düşünme biçimi, teknoloji kullanımı yerine strateji geliştirmeye öncelik verirken geleneksel HD yaklaşımından oldukça uzaklaşmaktadır. Denning (2017) ‘Geleneksel HD’ ve ‘Yeni HD’ olarak adlandırdığı bu iki yaklaşım arasındaki değişimi Çizelge 2.2 ile ifade eder.

Çizelge 2.2 : Eski HD ve Yeni HD (Denning, 2017; Türkçe'ye tercüme edilmiştir).

Geleneksel HD	Yeni HD
İşlevsel yazılımlar üretmede ve tasarlamada kullanılan zihinsel alışkanlıklar ve disiplindir.	Hesaplamalı aşamalarla ifade ederek çözüm üretmek üzere, problemleri formüle etmek için kullanılır.
Kapsamlı bir programlama pratiği yapmak HD'yi dizi halinde beceri olarak kazanmayı sağlar.	HD programlamayı sağlayan kavramsal çerçevedir.
Yazılım tasarım ve üretme becerileri; ayrımı, soyutlamanın etkin kullanımı, kişinin ihtiyaçlarına göre uyarlanmış gösterimlerin oluşturulması ve birleşimsel olarak patlayan durum analizlerinden kaçınmaktır.	Temsil, parçala-yönet, soyutlama, bilgi saklama, onaylama ve mantıksal muhakeme yapma gibi problem çözme kavramlar barındırır.
Bilimi yönlendirecek, kuram ve deney yönünden devrim niteliği taşır.	Bilim ve diğer alanlar için kullanışlıdır.
Algoritmalar, bir talimatı yerine getirmek için (soyut makineyi) sayısal bir modeli kontrol etmek üzere kullanılan yönlendirmelerdir.	Algoritmalar görevi yerine getirmede kullanılan tariflerin ifadesidir, sayısal modeller için farkındalık şartı yoktur.
Programlar algoritmalar ile sıkıca bağlıdır; programlar bilgisayar dilinde ifade edilen algoritmalar ve algoritmalar kesinliklerini hesaplama modellerinden alırlar.	Programlar algoritmalarla esnek şekilde bağlantılıdır; Algoritmalar, insan dahil her türlü verinin işlemcileridir ve bir algoritmayı programa çevirmek tercihe bağlıdır.
Bir alanda hesaplama tasarlamak kapsamlı alan bilgisi gerektirir.	HD hakkında eğitim alan birey, herhangi bir alanda karşılaştığı problem için çözüm üretebilir.
Son kullanıcılar algoritmaları takip edebilir ve mekanizmaya dair anlayış kazanmadan sonuç alabilir.	Adım adım işlemlere aşinalık kazanan bireyler, algoritmaları üretebilir ve belki de farkında olmadan hesaplamalı düşünebilir.
Farkındalık sahibi olmadan hesaplamalı bir göreve dahil olmak, hesaplamalı düşünme değildir.	Hesaplamalarla ortaya çıkan herhangi bir görevde bireyler farkında olmadan hesaplamalı düşünebilir.

HD'nin, eğitimin erken aşamalarından itibaren teşvik edilmesindeki temel amaçlarından biri, bu zihinsel aracı problem çözme veya problem tanımlama görevlerinde bireye alternatif bir çıkış yolu olarak kazandırmaya ve eğitim teknolojilerinin geliştirilmesiiçin ihtiyaç duyulan derin öğrenme deneyimlerini elde etmeye katkıda bulunmaktır.

Çünkü HD, “Bireylere hesaplanabilir sorunları tanımayı öğreterek, problem çözme sürecine etkili ve ustaca yaklaştırmaya yardımcıdır” (Czerkowski ve Lyman 2015; s:57).,

Ayrıca HD'nin,

- Karmaşıklıkla başa çıkma konusunda özgüven;
- Zor problemler karşısında istikrar;
- Belirsiz, açık uçlu problemlerle başa çıkabilme ve
- Ortak bir hedefe veya çözüme ulaşmak için başkalarıyla iletişim kurma
- gibi çalışma becerilerine olan katkılarından söz etmek mümkündür (Barr & Stephenson, 2011; s:22).

Bu kapsamda son on yıl içinde, Birleşik Devletler, Birleşik Krallık, Avrupa, Avusturalya gibi pek çok kıtada, HD'yi temel eğitim müfredatına erkenden kazandırmayı hedefleyen kurum ve kuruluşların sayısı ve etki alanında artış görülmektedir., The Australian National Curriculum, Danish Growth Council, British Royal Society, US National Research Council gibi isimler, HD'yi farklı yaş grupları ve eğitim alanlarında teşvik ederken; Türkiye’de bu sorumluluğu ilköğretimden lisans seviyesine kadar Millî Eğitim Bakanlığı (Bocconi ve diğ., 2016), lisans seviyesi ve üstünde ise Yükseköğretim Kurulu üstlenmiştir (TYYÇ, 2018).

Tüm bu gelişmeler ışığında ise, analog ortamdan dijital ortama dek çeşitlilik gösteren arayüz ve uygulamalar eşliğinde, temel bir beceri olarak eğitim müfredatındaki yeri sağlamlaşmış HD'nin pedagojik uyumu üzerine çalışmalar süregelmıştır. Bu çalışmalar, HD'nin bilişsel yönünü hedef alarak bireye daha iyi öğrenim deneyimleri kazandırmayı ve bilişim teknolojilerinin gelişimine katkıda bulunmayı hedeflemektedir. Nitekim HD'yi eğitim müfredatına geliştirilebilir/esnek bir beceri olarak kazandırma arzusunun,

- *Öğrenmeyi geliştirmek ve bilgisayar bilimi kavramlarının anlaşılmasını güçlendirmek için teknolojiyi nasıl kullanabiliriz?*
- *İlerlemeyi, öğrenme çıktılarını ve zaman içinde kalmayı ölçmek için teknolojiyi nasıl kullanabiliriz?*
- *Öğrenme hızında ve bilişsel yeteneklerde farklılıklara sahip olduğumuzu göz önüne alırsak, öğrenmeyi kişiselleştirmek için teknolojiyi nasıl kullanabiliriz?*

gibi bir dizi yeni araştırma konusunu da beraberinde getirdiği açıkça görülmektedir (Wing, 2017).

Bu duruma nazaran, National Science Foundation, British Royal Society, European Commission, Google for Education ve Microsoft Research Lab gibi pek çok kurum ve kuruluşun çevrimiçi/çevrimdışı platformlarda sundukları öğrenim deneyimlerini referans alarak, kendi HD tanımlarını ortaya çıkardıkları görülmektedir. Derinlik ve nitelik yönünden çeşitlilik taşıyan bu tanımlarda yer verilen eylem ve becerileri ise Çizelge 2.3'teki gibi sıralamak mümkündür.

Çizelge 2.3 : HD kavramları ve karşılıkları (uyarlamadır; Url-4; Automation).

HD Kavramları	Karşılıkları
Soyutlama: (Abstraction)	ana fikri ifade etmede rol oynayan bilgiyi tanımak ve ortaya çıkarmak
Algoritmik Düşünme (Algorithmic Thinking)	benzer problemleri çözmek için veya bir görevi tamamlamak için seri talimatlar dizisi oluşturmak
Veri Analizi (Data Analysis)	örüntü arayarak veya öngörüler geliştirerek veriden anlam çıkarmak
Veri Toplama (Data Collection)	bilgi toplama
Veri Temsili (Data Representation)	grafik, tablo, kelime veya grafiklerle uygun ifade biçimlerini organize etmek
Eşleştirme (Parallelization)	ortak hedefe daha etkili ulaşmak amacıyla tanımlanan büyük görevde yer alan küçük görevlerin eş zamanlı olarak yapılması
Genelleme (Generalization)	tahmini sonuçları test etmek üzere modeller, kurallar, ilkeler ve kuramlar oluşturmak
Örüntü Tanıma (Pattern Recognition)	veride yer alan örüntüleri, eğilimleri ve tekrarları gözlemlemek,
Benzetim (Simulation)	gerçek hayat sürecini taklit eden model geliştirmek.
Hata Ayıklama (Debugging)	hataları veya arızaları yakalamak.
Problem Çözme (Problem Solving)	problem için çözüm üretme süreci
Ayrıştırma (Decomposition)	veriyi, süreci veya problemi kontrol edilebilir küçük parçalara ayırmak.
Özişler (Automation)	tekrarlayan görevler için makinelerden yararlanmak

Bir yandan bu kavramların, HD'yi disiplinlerarası çerçevedeyorumlamaya oldukça yardımcı olduğunu belirtmek mümkün iken; diğer yandan, bu kavramların ifade ve ilişkilendirilme biçimlerindeki çeşitliliğin, HD'nin pedagojik uyumu konusunda birtakım zorlukları beraberinde getirdiğini söylemek mümkündür. Çizelge 2.3'e bakıldığında bazı eylem ve becerilerin birbiriyle yakınlık içinde olduğu; bazılarının ise disiplinler bakış açısı altında öne çıktığı görülmektedir.

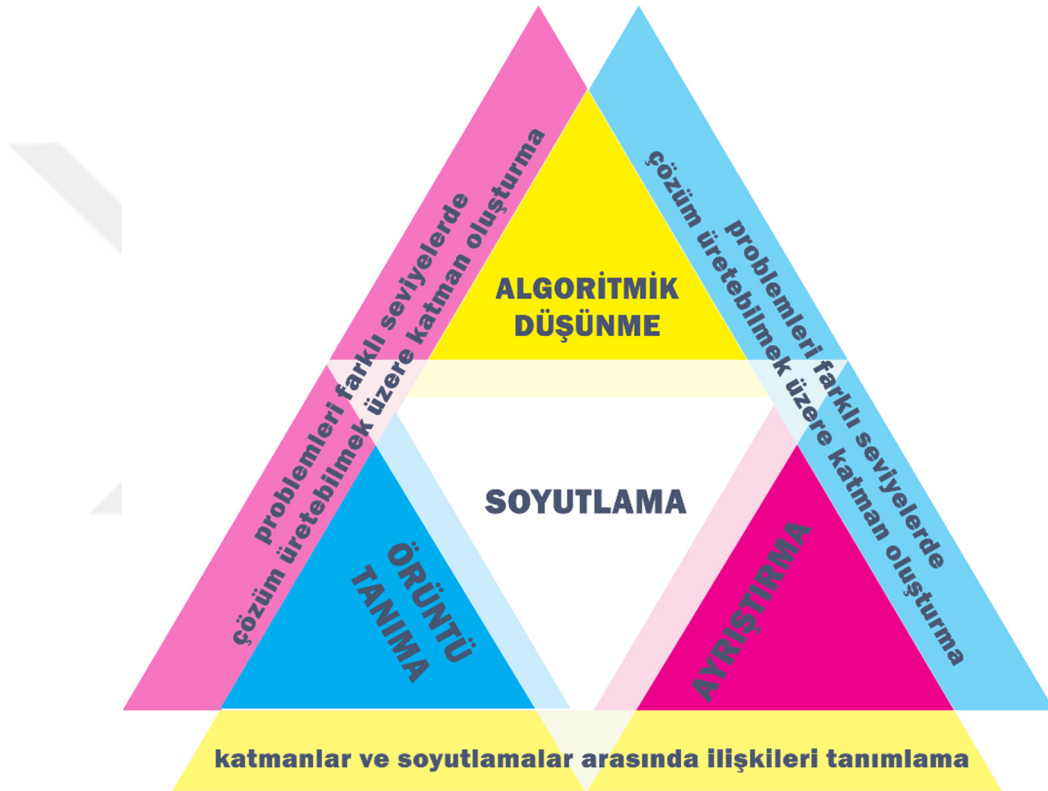
Çizelge 2.4'te farklı kurum, kuruluş ve eğitimcilerin HD tanımlarında yer vermiş oldukları eylem ve becerilerin kullanım sıklıkları ve içeriklerinin karşılaştırmalı olarak dikkate alındığında, bazı eylem ve becerilerin tüm tanımlarda yer aldığını, bazılarının ise içerik ve ifade yönünden çeşitlilik taşıdığı açıkça görülmektedir.

Nitekim Mark Guzdial'ın (2008) ifade ettiği üzere "Bilgisayar Bilimlerinden diğer alanlara HD için yol çizmek, bu düşünme biçimini acemi veya alan dışından kişilerin ihtiyaçlarına uygun biçimde uyarlamayı gerektirmektedir". Fakat bunu yaparken terimlerin referans oluşturduğu bazı düşünme biçimlerinin, hesaplama yöntemi olarak potansiyel taşınmasına rağmen "hesaplamalı" görünmeyebilir (Hu, 2011) olduğu da unutulmamalıdır. Bir zihin aracı olarak HD, farklı düşünme biçimleri ve becerilerin anlamlı ilişkiler çerçevesinde eylemsellik kazanmasıdır. HD'nin bilişsel süreci, soyutlamaları tanımlamak; çok sayıda soyutlama katmanıyla çalışmak ve farklı katmanlar arasındaki ilişkileri anlamak ve oluşturmak üzere üç aşamadan oluşur (Wing, 2008; s:3718).

Çizelge 2.4 : Yeni yaklaşımı çerçevesinde yer verilen HD tanımlarında yer alan kavramların dağılımı (2006-2018).

HD Kavramları	Eğitim için Google (Url-5)	Ulusal Araştırma Konseyi (NRC, 2010)	Uluslararası Teknoloji Toplumu (ISTE, 2011)	Avustralya Müfredatı, Değerlendirme ve Raporlama Kurumu (ACARA, Url-6)	Selby ve Woollard (2013)	Wing (2008)	Kalelioglu, Filiz; Gülbahar, Yasemin; Kukul (2016)	Avrupa Birliği Araştırma Raporu (Bocconi ve diğ., 2016)	Microsoft Araştırma Konseyi (Url-1)	Barefoot Computing, (Url-7)
Soyutlama	X	X	X	X	X	X	X	X	X	X
Algoritmik Düşünme	X	X	Algoritma & Prosedürler	X	-	X	X	Problem Çözme	X	X
Veri Analizi	X	X	X	-	-	-	X	-	-	(Logic) Mantık yürütme
Veri Toplama	X	-	X	-	-	-	X	-	-	-
Veri Temsili	X	Veri modelleme	X	-	-	-	-	-	-	-
Eşleştirme	-	-	X	-	-	-	X	-	-	-
Genelleme	Örüntü Genelleme	-	-	-	-	-	-	-	-	(Evaluation) Değerlendirme
Örüntü Tanıma	X	X	X	X	X	X	X	X	X	X
Benzetim	X	-	X	-	-	-	X	-	-	-
Hata Ayıklama	X	-	-	-	-	-	-	-	-	X
Problem Çözme	X	-	-	Algoritmik Düşünme	-	-	-	Algoritmik Düşünme	-	-
Ayrıştırma	X	X	Problem Ayrıştırma	X	X	X	X	X	X	X
Özişler	X	X	X	-	-	-	-	-	-	-

Bu çerçevede Çizelge 2.4'e yeniden bakıldığında, literatürde yer verilmiş olan HD tanımlarının hemen hepsinde soyutlama, ayırıştırma, örüntü tanıma ve algoritmik düşünme kavramlarının birlikte kullanılması oldukça dikkat çekicidir. Bir zihin aracı olarak HD, bu dört temel değer birbirleriyle ilişkilendirilmesine ihtiyaç duymaktadır (Şekil 2.3). Nitekim eğitim teknolojileri literatüründe bireyin HD becerilerini ölçmeye ve değerlendirmeye alan araştırmaların da bu dört bilişene ağırlık gösterdiğini söylemek mümkündür (örnek: Moreno-Leon, Roman-Gonzalez, & Robles, 2018).



Şekil 2.3 : Hesaplamanın Zihinsel Araçları; Wing'in (2008) tanımından uyarlamadır.

Ancak HD'yi bir zihin aracı olarak farklı disiplinlerde teşvik etmek için bu dört bilişsel değere eğitim müfredatı içinde yer vermiş olmak tek başına yeterli değildir. Bu düşünme biçimini farklı disiplinler içinde de bir zihin aracı olarak sunabilmek ve bireye HD'nin bilişsel değerlerini esnek beceriler olarak kazandırmak için hem bileşenlere dair kavramsal yaklaşımın mesleki çerçevede dikkate almaya hem de bu çerçevede öğrenim deneyimleri oluşturmaya ihtiyaç duyulmaktadır. Aksi durumda, bireyin mesleki çerçevesi dışında kazandığı deneyimi, farklı durumlara uyarlamada zorluk yaşaması olasıdır (Guzdial v.d., 2010).

Bu kapsamda bir sonraki bölümde, çağdaş mimarlık eğitimindeki HD'nin rolü, öğretim kültürü ve bilişsel değerleri yönünden pedagojik uyumu ele alınmaktadır.

2.3 Mimarlık Eğitiminde Hesaplamalı Düşünme ve Pedagojik Uyumu

Bir tasarımcı için dijital araçlarda yetkinlik sahibi olmak; en az eskiz yapmak, maket yapmak gibi geleneksel tasarım ve temsil araçlarında yetkinlik sahibi olmak kadar önemlidir. İlk örneklerine kıyasla, CAD/CAM/CAE araçları ve yapay zekâ uygulamalarının bu disiplin içindeki kullanım alanı oldukça genişlemiş ve çeşitlilik kazanmıştır. Günümüz tasarımcısı için dijital araçlarla tasarlamak ve bu araçları özgün biçimde kullanmak, analog ortamda bir ressamın fırça ve teknik kullanımı kadar öznelendirilebilir becerilerdir (Reas & McWilliams, 2011). İyileştirilebilir kullanıcı arayüzleri, nesne tabanlı ve görsel programlama dilleri ile geliştirilebilir araç ve programlar; mimarlıkta temsil, tasarım ve üretim sürecinin yadsınamaz nitelikteki parçalarıdır. Bu nedenle, hesaplama ve kodlama kültürüne yöneldiğimiz bu paradigma ile mimarlık öğrencileri eğitimlerinin ilk aşamalarından başlayarak birer dijital okur-yazar/sanatçı olma yolunda ilerlediğini söylemek mümkündür. Ayrıca son on yıldır “dijital tasarım teknolojilerini kullanma yetkinliği,” Birleşik Devletler Mimarlık Akreditasyon kurulunca değerlendirme kıstaslarından biri olarak tanımlanmıştır (NAAB, 2009).

Mimarlıkta da HD'nin öncelikli rolü, bilişim teknolojilerinin kullanımında karşımıza çıkar. Geleceğin tasarımcısı için, dijital araçlarda yetkin olmak, bahsedilen etkileşimli ve iş birlikçi platformlarda etkin ve verimli çalışabilmek, ‘eğitilmiş bir göz’ kazanmakla mümkündür ve bilişim teknolojilerinde görme ve davranış biçimindeki ihtiyaç duyulan farkındalığı kazandırabilecek olan hesaplamalı düşünme yaklaşımıdır. “Bilgisayarlar, dijital üretim makineleri ve benzeri araçlarla yaratıcı ve etkili çalışmak, akademik ve profesyonel davranışlar için bir dizi yeni iş akışı ve uyarlamayı da beraberinde getirmektedir” (Doyle & Senske, 2016; s:207). Bu anlamda HD, bir tasarımcının bu araçlarla çalışmasına yardımcı olabilecek bir zihin aracıdır (Senske, 2014). Bu zihinsel araç, tasarımcının karşılaştığı bir problemi eğitilmiş bir göz ile formüle etmesini sağlar. Kazandırdığı soyutlama, örüntü tanıma, ayrıştırma ve algoritmik düşünme becerileri ile, bireyin bilişim ortam ve araçlarıyla etkin ve verimli şekilde çalışmasına katkıda bulunur.

Diğer yandan, mimarlık eğitiminde HD'nin ne olduğu ve bilişim teknolojileri ile nasıl ilişkilendirilmesi gerektiği hakkında oldukça sınırlı ve geleneksel görüşlerin varlığından söz etmek mümkündür. Gül ve diğ.'nin (2013) Türkiye'deki mimarlık okullarında bilişim teknolojilerinin eğitimde kullanımı üzerine yapmış oldukları kapsamlı çalışmada, oldukça az sayıda okulun müfredatında teknoloji kullanımına yer verdiğini saptamış ve bilişim teknolojilerinin eğitimi kapsamında bir altyapı eksikliğini gözler önüne sermiştir. Araştırmaya göre, çoğu mimarlık okulu bilişim teknolojilerin kullanımı katı nesne modelleme uygulamaları ile sınırlıdır (Gül ve diğ., 2013).

Öncelikle ne hesaplama ne de hesaplamalı düşünmenin mimarlığa yabancı kavramlardır. Ancak “hesaplamalı düşünmeyi öğrenmek, dijital tasarım yazılımını öğretme yöntemlerine kıyasla daha kalıcı bir eğitim değerine sahiptir (Senske, 2011; s:94). Temel bilişsel değerleriyle HD hem görme hem de davranış biçimleri bağlamında tasarımcının gelişimine katkıda bulunur. Çünkü “HD, problemler hakkında düşünmektir ve HD ile problem çözmek, teknolojinin gelişmesine ayrıca destek sağlar” (Url-7). Bu nedenle yalnızca öğrenciler değil, eğitimciler ve teknoloji geliştiriciler için de mimarlık eğitiminin bu kavrama müfredat içinde teknik bir yeterlikten fazlası olarak değer vermesi önemlidir. Çünkü Alan Blackwell'in (2002) de belirttiği üzere, “tüm hesaplama biçimleri -her türlü yazılım, programlama dili ve cihazlar- özünde farklı hesaplama türleri veya arayüzleri olarak kabul edilmelidir.” Hiç şüphesiz, bu kavramlar, mimarlık eğitiminde teknoloji kültürünü ve yakın geleceğini şekillendirmede stratejik önem taşımaktadır.

Bu çerçevede, yakın zaman içinde söz konusu altyapı eksikliğini iyileştirmeye yönelik çalışmalardan bahsetmek mümkündür. Mimarlık öğrencilerinin bilişim teknolojileri ile tanıştırılma biçimleri ve ihtiyaç duydukları HD'nin öğrenim deneyimi, eğitimde farklı ekollerle karşılanmaktadır. Birleşik Devletler'den University of North Carolina Charlotte (Senske, 2011), ülkemizden İstanbul Teknik Üniversitesi (Url-8) gibi kimi mimarlık okulları eğitim müfredatları içinde radikal değişiklikler göstererek; tasarımda hesaplamalı yöntem, araç ve teknoloji kullanımına zorunlu eğitimin bir parçası olarak yer vermiş iken; kimi okullarsa bu ihtiyacı temel tasarım, seçmeli dersler veya proje dersleri altında yer verdikleri farklı uygulamalarla çözüm aramıştır.

M2					1st Year		2nd Year			
					Comp. Methods		7103	7102		
M1	1st Year				2nd Year			3rd Year		
	5050/ 6111		6112		Comp. Methods		7103	7102		
UG	1st Year		2nd Year		3rd Year		4th Year		5th Year	
	Fall	Spring	Fall	Spring	Fall	Spring	* Fall	Spring	Fall	Spring
	1101	1102/1602	2101	2102	Computational Methods	3102 *	4101	4102	Computational Practices	4104 *
IDEAS	Introduction to Computational Processes		Basic CAD Techniques	Topology	Adv.Topology Strategic Thinking & Processes	Strategic Thinking & Processes	TOPICAL	TOPICAL	Strategic Thinking & Processes in Practice	Strategic Thinking & Processes
SKILLS	2D Graphics Digital Media Concepts Interfaces		2D CAD	Basic 3D Modeling	Advanced 3D Modeling Basic Scripting Basic Parametric Design				Adv. Scripting Adv. Parametric Design Research Methods and Writing	Advanced Information Modeling
AWARENESS					EXPLICIT UNDERSTANDING				MASTERY	

Şekil 2.4 : UNC Charlotte Mimarlık Okulu, Bilgisayar ve Hesaplama Programı, (Senske, 2011).

Ders Bilgileri (Course Catalog)

Kod (Code)	Ders Adı (Course Name)	Dili (Language)	Türü (Type)	
MIM 120E	Mimaride Hesaplamalı Tasarım Araçları ve Yöntemlerine Giriş <i>Introduction to Computational Design Tools and Methods in Architecture</i>	İngilizce (English)	Zorunlu (Compulsory)	
Kredi (Local Credits)	AKTS (ECTS)	Ders (Theoretical)	Uygulama (Tutorial)	Laboratuvar (Laboratory)
2	4	1	2	0
Dersin Önşartı ve Sınıf Kısıtı (Course Prerequisites and Class Restriction)				
Ders Önşartı (Prerequisites)		Yok/None		
Sınıf Kısıtı (Class Restriction)		Yok/None		
Ders Tanımı (Course Description)				
Bu zorunlu ders kapsamında hesaplama kavramıyla ilgili genel bilgiler verilmekte ve mimari tasarımdaki çeşitli uygulamalarına ait geniş bir çerçeve çizilmektedir. Ayrıca, öğrencilere çeşitli dijital tasarım, fabrikasyon ve üretim teknolojilerinin mimari tasarımdaki uygulamalarına dair giriş seviyesinde bilgi verilmekte ve yine giriş seviyesinde geometric modelleme yazılımı (Rhinceros) ve algoritmik düşünce yazılımı (Grasshopper) ve laser kesici kullanımına yönelik uygulamalı bilgi ve yöntemler de aktarılmaktadır. Bu amaçlarla, haftalık planda da görülebileceği gibi, kuramsal derste geniş bir içerik aktarılmakta ve uygulamalı ders saatlerinde de, Rhinceros ve Grasshopper deneyimi kazandırılmaktadır.				
<i>This is a compulsory unit of study which aims at introducing students to general principles of computation and its wide range of applications in architectural design. It also aims to introduce the students to the various digital design, fabrication and manufacturing technologies utilized in architectural design practices and make them develop a beginner level knowledge and experience in a suite of these technologies which involve a geometric modeling application (Rhinceros), together with an algorithmic modeling plug in (Grasshopper) and laser cutting as a method for digital fabrication. With these objectives, as can be seen in the weekly plan, the course covers a wide range of topics to be delivered in lecture hours and tutorials and hands on working with Rhinceros and Grasshopper during the lab hours.</i>				

Şekil 2.5 : İTÜ Mimarlık Fakültesi, imaride Hesaplamalı Tasarım Araçları ve Yöntemlerine Giriş dersi katalog formu, (Url-8, 2019).

Ayrıca bazı eğitimcilerin ‘Tasarı Geometri’ gibi müfredattan çıkarılmış/çıkarılmak üzere olan derslerin içeriğini hesaplmalı tasarım yaklaşımları ve araçlarına bırakmak üzere yeniden müfredata kazandırma girişimleri de mevcut çözüm önerileri arasında yer almaktadır (Ko & Steinfeld, 2018; s:4). Bu çerçevede, bilişim teknolojilerinin eğitim müfredatında yer alışı biçimleri oldukça çeşitlilik göstermektedir.

Ancak bu çeşitlilik içinde, eğitimeiler arasında bilişim teknolojilerinin kullanımı ve zihinsel bir araç olarak HD'nin öğretim kültürüne dair fikir birliğinin oluşmadığı açıkça görülmektedir. Dolayısıyla mimarlık eğitiminde HD'nin pedagojik uyumu için bu yönden bir boşluğun varlığından söz etmek mümkündür. Brian Johnson'ın (2017) belirttiği üzere "Dijital veya kağıtsız stüdyo deneylerine yer veren birkaç okulun haricinde, dijital analiz ve medya seçeneklerinin ortaya çıkışı, her biri geleneksel strüktürün üstünde bir seçenek olan yeni araçlarla beceri kazanmaya odaklanan özel ders içerikleriyle ele alınmaktadır."

"Mimarlık eğitimi içinde en az değinilen konulardan biri insan ve makine düşüncesinin rolleri üzerine yapılan ciddi tartışmalardır ve bu alanda yapılan çalışmaların çoğu tasarım ve üretim otomasyonunun yaygın kullanımına değinmektedir" (Senske, 2014). Bu bağlamda, mevcut eğitim müfredatı içinde HD'nin öğretim kültürünü iyileştirmeye yönelik araştırmalara ihtiyaç duyulduğunu söylemek mümkündür. Nitekim Wing'in tanıştırdığı yeni HD yaklaşımından günümüze kadar (2006-2019), WebofScience veritabanında, "computing" veya "computational thinking" veya "computation" "design pedagogy" veya "design education" veya "architecture education" konuları ile "education" veya "pedagogy" veya "education" veya "architecture education" anahtar kelimeleriyle yapılan bibliyografik taramanın sonuçları, 'Mimarlık Eğitiminde Hesaplamalı Düşünmenin Öğretim Kültürü'nü ele almış araştırmaların sayı ve nitelik açısından oldukça az olduğunu göstermektedir (Şekil 2.6).



Şekil 2.6 : WebofScience veri tabanında hesaplamalı düşünmenin mimarlık/tasarım eğitimi ile ilişkilendirildiği yayınların sayıları ve konu alanları.

Bu tarama sonuçlarını

- Mimarlık öğrencileri HD'ye ne zaman ve neden ihtiyaç duyar?
- HD bir tasarımcı adayının öğrenme deneyimlerine ne yönlerden katkıda bulunur?
- Tasarımcı bakış açısı HD'nin temel bilişsel değerleri ile ne derece uyumludur?
- HD'nin hangi araç ve yöntemlerle kazandırılması görsel bir düşünür için avantaj sağlar?

sorularıyla mercek altına aldığımızda, Senske'nin görüşünü destekler biçimde, mevcut araştırmaların ağırlıklı olarak hesaplama modelleri, program ve araç kullanma becerilerine teknik yönden ağırlık gösterdiği görülmektedir. “Son yirmi yıldır yeni dijital tasarım ve uygulama alanları durmaksızın ortaya çıkmaktayken; hesaplama giriş adı altında, her birinin farklı bir konu seti önerdiği (doğadan esinlenen tasarım, dijital üretim, beliren sistemler ve veri görselleştirme dahil olmak üzere) bir dizi dijital-esinlenmiş hareketler ve estetik stiller ortaya çıkmış ve kaybolmuştur” (Ko ve Steinfeld, 2018; s:4). Çoğu araştırmada, bilişim araçlarının kullanımı tasarımda yaratıcılık, yaparak öğrenmek gibi biçimsel (formalization) konulara ağırlık göstermektedir. Araç ve kullanıcı etkileşimine ise farklı tasarım araç ve ortamlarının karşılaştırmalı örnekleri çerçevesinde yer verilmektedir.

Bu kapsamda, HD'nin mimarlıktaki öğretim kültürünü ortaya çıkaran araçlar ve müfredata yansıma biçimlerini,

- Programlama dillerinin pratiği ile bilgisayar okuryazarlığı kazandırmak;
- Hesaplamalı tasarım araçları ve modelleri aracılığı ile deneyim kazandırmak;
- Bilişsel değerlerin araçsallaştırılması ile beceri kazandırmak;

olmak üzere üç ana başlık altında toplamak mümkündür. Her bir yöntemin kendine özgü avantajı olsa da bu yöntemlerin müfredata yansıma ve tasarımcı yaklaşımına uyarlanış biçimlerine dikkat edilmelidir. Çünkü, bu düşünme biçimini bireye bir zihin aracı olarak kazandırmak, hem bilişsel değerlerin ihtiyaç duyduğu kavramsal yaklaşımı hem de bu değerleri birlikte kullanmaya yönelttiği öğrenim deneyimlerini birlikte kurgulamayı gerektirir.

Sunulan öğrenim deneyiminde yer verilen araç ve ortam her ne olursa olsun, önemli olan bireye soyutlama, örüntü tanıma, ayırıştırma ve algoritmik düşünme çerçevesinde anlamlı ilişkiler kurdurabilmektir. Hesaplamalı düşünme yalnızca birtakım bilişsel değerler hakkında bilgi edinmeyi değil; aynı zamanda bu bilgiyi farklı problemler ve durumlara uyarlayabilmeyi gerektirir. Bu nedenle bilişsel değerlerin kazanımı konusunda değerlendirme yapabilmek oldukça önemlidir.

Bilişim teknolojilerinin öğretiminde mevcut ekolumüzü HD'nin temelini oluşturan kavramlarla ilişkilendirmek, HD'nin pedagojik uyumu çerçevesinde önemli adımlar atmaya yardımcı olacaktır. Ancak bu ilişkilendirmede, tasarım eylemini hedef alan bir dizi aksiyomatik uygulama yerine, tasarımcı bakış açısını temel alan özelleşmiş öğrenim deneyimleri ile yer verilmelidir. Bu bağlamda hem bilişim araçlarını etkin ve verimli kullanmak hem de bu araçların gelişimine katkıda bulunmak mümkün olacaktır.

Bir zihin aracı olarak HD'yi teşvik eden öğrenim deneyimlerinde,

- Farklı ifade biçimleri üzerinden soyutlama yapabilmeye;
- Yapılan soyutlamaların niteliğini ve düzeyini kavramsal ve prosedürel yönden tanımlayabilmeye;
- Problem tanımaya veya çözüm üretmeye yönelik yapılan soyutlamalarla genellemeler yapabilmeye;
- Genellemeler arasında anlamlı ve kurallı ilişkiler oluşturabilmeye yer verilmelidir.

Bu çerçevede, mevcut mimarlık eğitiminde yer alan HD'nin öğretim kültürü ve kavramın pedagojik uyumundaki boşluğun nedenlerinden bazılarını, bu düşünme biçiminin tasarım eğitimi ile özdeşleştirilmiş veya bağlamdan koparılmış öğrenim deneyimleri ile sunulması ve bu öğrenim deneyimlerinde tasarımcı seviyesinin dikkate alınmaması olarak göstermek mümkündür.

Mevcut literatürde, mimarlık öğrencilerinin, hesaplamalı yöntem ve modelleme araçları ile deneyimlerinde, hesaplamalı düşünmeyi bir zihin aracı olarak ne kadar etkin veya verimli kullandıkları tartışılmamaktadır.

Sunulan öğrenim deneyimlerinin pek çoğunda seçilen araç ve yöntemlerin ele alınış biçimi, bilgisayar okuryazarlığı seviyesinde gerçekleşmekte; tasarımcı bakış açısının ve düzeyinin HD bileşenleri üzerindeki etkisi dikkate alınmamaktadır. Bireyin bu bileşenleri birbiri ile nasıl ilişkilendirdiği ya da hangi bileşen yönünden ne kadar gelişim gösterdiği, neye ihtiyacının olduğu göz ardı edilmektedir.

Sonuç olarak, HD'yi teşvik etmek için, temel beceri ve eylemlerinin görsel bir düşünür için kavramsal yönden taşıdığı farklılıklara dikkat edilmelidir. Sunulan öğrenim deneyimleri içerisinde, tasarımcı bakış açısını dikkate alan ve tasarım uzmanlığı (eğitim seviyesi) yönünden engel teşkil etmeyecek uygulamaların tercih edilmesi gerekir.

Ayrıca, sunulan öğrenim deneyimi, en azından bu dört bilişsel beceriyi belirli bir problem ya da durum karşısında birlikte kullanmaya teşvik edecek şekilde kurgulanmalıdır. Bunun içinse öncelikli koşul, tasarımcıya farklı düzeylerde soyutlama yapabilecek deneyimi kazandırmaktır. Bu tür soyutlama deneyimleri tasarımcıların görme ve davranış biçimleri arasında ilişki kurabilmesine yardımcıdır. Nitekim, farklı düzeydeki soyutlama yaklaşımları, özellikle görsel bir düşünürün HD sürecini belirgin şekilde etkilemektedir. Bilişim araçlarıyla çalışan görsel düşünürler, soyutlama kavramına hem kavramsal hem de prosedürel anlamda ihtiyaç duymaktadır.

Eğitimi süresince görsel düşünmeye teşvik edilen tasarımcılar, soyutlama becerilerini kavramsal düzeyde kullanmaya eğilim gösterirler ve bu durum hesaplamalı düşünmek için ihtiyaç duydukları prosedürel yönden soyutlama yapmalarına veya beceri kazanmalarına engel teşkil edebilir. Bu bağlamda mesleki eğitimi çerçevesinde bir tasarımcının kazandığı soyutlama becerisinin bir mühendisin kazandığından oldukça farklı düzeyde gelişim gösterdiğini söylemek mümkündür. Bu durum da tasarımcıların eğitim sürecinde, prosedürel soyutlama becerilerini geliştirmeye yönelik deneyim kazanmaya ihtiyaç duyduklarını ortaya çıkarmaktadır.

3. GÖRSEL BİR DÜŞÜNÜRÜN HESAPLAMALI DÜŞÜNME YAKLAŞIMI

Mimarlık eğitimi, ilk yıldan itibaren yoğun olarak yer verdiği çizim, eskiz, modelleme gibi tasarım ve üretim araçları ile bireyin görsel düşünme ve görsel iletişimini pekiştirmeyi hedefler. Bu doğrultuda, tasarımcı adayı görsel bir düşünür olma yolunda hızla adım atar. “Görsel düşünme, bir dizi dikkat eyleminden, göz hareketlerini yönlendirmekten ve örüntü bulma bağlantılarımızı ayarlamaktan ibarettir. Bu dikkat eylemleri birer görsel sorgulamadır ve görsel sorgulamanın nasıl çalıştığını anlamak bizi daha iyi tasarımcılar yapabilir” (Ware, 2008; s:3). Ancak bu konuda bir tasarımcı kimliğini diğer disiplinlerden ayıran, tasarımcının görsel düşünme sürecinde algısal ve sezgisel yaklaşıma ağırlık göstermesidir.

Tasarımın yaratıcı aşamalarında, tasarımcılar görme biçimlerini tekrar tekrar değiştirir (Schön, 1987). Aynı görsel imgelerle çalışan bir tasarımcı ve bir bilimci karşılaştırıldığında, tasarımcının sahip olduğu soyutlama yaklaşımının, bilimcinin soyutlama yaklaşımı kadar algısallıktan arındırılmış düzeyde gerçekleşmeme ihtimali her zaman vardır. Bir sanatçı ya da tasarımcı olarak görsel bir düşünür, tasarımı üzerinde herhangi bir değişiklik yapmadan önce, olası tasarım değişikliklerini test etmek için zihinsel olarak basit kalıplar ekleme potansiyeline sahiptir (Ware, 2010; s: 170). Lawson'a (2006) göre, "Mimarlar tasarım çalışması yoluyla öğretilir; onlardan sorunları anlamaları ya da durumları analiz etmeleri istenmez. Buna karşılık, bilimcilere teorik olarak öğretilir. Bilimin açık bir yöntemle ilerlediği öğretilir ve bunlar başkaları tarafından kopyalanabilir." Başka bir deyişle, bilimciler ve mimarlar bir soruna yaklaşırken farklı stratejiler izlerler. Bilimcilerin aksine, mimarlar çözüm odaklı stratejiler yaparak soyutlama deneyimi kazanırlar ve öğrenim deneyimlerindeki bu fark, zamanla soyutlama becerilerinin kullanımına yansır.

Bu nedenle soyutlama, genelleme, örüntü tanıma, ayırttırma ve algoritmik düşünme olan HD'nin temel bilişsel değerleri, mimarlık için yabancı olmasa da bu değerlerin eğitim müfredatında ele alınış biçimleri, mühendislik bilimlerinde ele alınış biçimlerinden farklı olduğu söylenebilir.

Mühendislik disiplinlerinin aksine, sanat ve zanaat ağırlıklı geçmişi ile mimarlık eğitimi, bu kavramları daha sezgisel yaklaşımlarla ve somutlaşmış (embodied) öğrenim deneyimleriyle öğrencisine sunar. Dolayısıyla, öğrenim deneyimlerinde ortaya çıkan bu nüansın, HD'nin öğretim kültürü ve pedagojik uyumunda dikkate alınması oldukça önemlidir. Hu'nun (2011) vurguladığı gibi, “Birey aynı düşünme yeteneğini çok farklı bir öğrenme deneyimiyle kazanmış olabilir ve bu nedenle günümüzde hesaplamalı düşünmenin önemini vurgulamak, felsefi bir meydan okumaya dönüşebilir ve bu şekilde daha modern bilgisayarlar ortaya çıkmamışken var olan hesaplamalı düşünme ürünlerini ortaya çıkarmaya yardımcı olabilir (s:20).”

Nitekim, araştırmının bu bölümünde de HD altında yer alan temel bilişsel değerlerin tasarım disiplini altındaki kavramsal karşılıkları ve mimarlık eğitimine yakınsamasından bahsedilmektedir. Literatürdeki çalışmaların aksine bu çalışmada, HD'nin bilişsel değerlerinin tasarımcı bakış açısıyla eylemselliği, soyutlama ve soyut düşünme üzerinden konu edilmektedir. Bu doğrultuda, bireyin tasarım eğitiminde soyutlama, örüntü tanıma, ayırıştırma ve algoritmik düşünme için sunulan kavramsal çerçevenin, HD bağlamında tanımlanan kavramsal çerçeve ile örtüştüğü ve ayrıştığı yönleri değerlendirilmektedir.

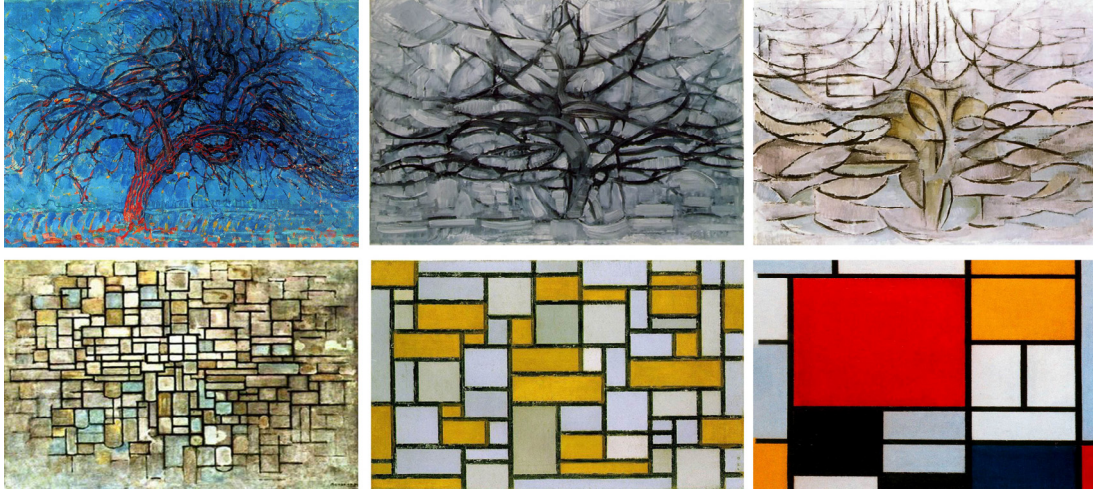
3.1 Görsel Düşünürler İçin Temel Tasarım Kavramları: Soyutlama, Örüntü Oluşturma ve Ayırıştırma

Soyutlama veya soyut düşünme hem Bilgisayar Bilimleri (Wing, 2008) hem de Tasarım (Goldschmidt, 2011) için temel kavramlardandır. Ancak bir mühendis için soyutlamanın kavramsal karşılığı, bir sanatçı ya da tasarımcı için her zaman benzerlik göstermez. Bir bilgisayar bilimcisi gözüyle Wing (2008) HD'de soyutlamanın görevini “bir çözümü, bir sorundan diğer benzer sorunlara genelleme ve aktarma yeteneği” olarak ifade eder ve şu cümleleri ekler: “soyutlama son derece geneldir, çünkü semboliktir... veri içine gömülü veya yayılmış bilgiyi çıkarmak veya veri temsil etmek ve işlemek üzere kullanılır” (ss:3717-3720). Bu noktada, Wing'in bu ifadesi fizik bilimci ve filozof John Locke'un soyutlama yaklaşımı ile benzerlik gösterir (Sengupta & Kinnebrew, 2013). (Locke, 1841) için soyutlama, “varlıklardan alınan fikirlerin, aynı türünün temsilcilerine dönüşmesidir.” Dolayısıyla, bu tür yaklaşımlar için daha çok bilgi-işlemsel (procedural) seviyede bir soyutlamadan bahsetmek mümkündür.

Algı bilimcisi Rudolf Arnheim, (2004) ise Locke'un sunduğu bu soyutlama anlayışını, herhangi algısal teminattan arınmış ve duysal deneyime dayanmayan, kelimelerle ifade edilmiş bir soyutlama olarak nitelendirmiştir (s:154). Bu çerçevede sembolik ve bilgi-işlemsel soyutlama ile kazanılan öğrenim deneyimi, görsel düşünürün ihtiyaç duyduğu algısal yaklaşımı ifade etmede oldukça yetersizdir. Dolayısıyla, HD'nin tanımında yer alan soyutlamanın kavramsal karşılığının, bir sanatçı ya da tasarımcının kazandığı soyutlama ile her zaman benzerlik göstermediğini belirtmek mümkündür.

Sanat ve tasarım alanında soyutlama yapmak, “nesnelerden elde edilen fikirleri alarak ve bunları diğer tüm varlıklardan, gerçek varoluş koşullarından, zamandan, yerden ya da eşlik eden diğer fikirlerden ayırmaktır” (Arnheim, 2004; s:154). Kısaca, bir tasarımcının soyutlaması algıyla ve deneyimle yakından ilişkili ve daha çok kavramsal (conceptual) düzeydedir. Nitekim Michael Tovey'in (2015) ifade ettiği gibi tasarım, tasarımcıya göre değişiklik gösteren özel veya özelleşmiş düşünce biçimlerinin birlikteliğidir.

Görsel bir düşünür için soyutlamalarla uğraşmak, çizim, eskiz veya modellemenin ötesinde çok sayıda form ve temsil biçimleri arasında ilişki kurmayı gerektirir ve bu ilişkilerin dönüşümü içinde algısal müdahalelere sıkça rastlanır. Bu görsel düşünme biçimi tasarımcıya özgü bilme biçimi olarak önemlidir (Tovey, 2015; s:58). Ek olarak, ileri derecede soyutlama her zaman kendisini fiili varoluş zenginliğinden ayırma riskine sahiptir (Arnheim 1974, s:147). Goldschmidt (2011) soyutlamanın sabitlenmeyi tersi yönde etkilediğini savunur. Ancak bu becerinin tasarımcı seviyesine göre çeşitlilik göstereceği unutulmamalıdır. Çünkü bir açıklama veya tasvirden anlam çıkarmak, kısmen indirgeyici bir süreçtir; olası anlamların alanı, anlamları ifade etmenin yollarından daha büyüktür; yanlış kullanım, yanlış yorumlamalar ve yanlış anlamalar başarılar kadar kaçınılmazdır (Tversky, 2011). Bu nedenle, acemi bir tasarımcının görsel imgeler üzerinden kavramsal düzeyde gerçekleştirdiği soyutlamanın seviyesini nitelendirmek aslında oldukça zordur. Söz konusu algısal yaklaşımlar, kavramsal düzeyde yapılan soyutlamanın nicel yönden derecelendirilmesini imkânsız hale getirir. Bu duruma başarılı bir örnek olarak Piet Mondrian'ın çalışmalarını göstermek mümkündür. Şekil 3.1'de sol üst köşede verilen “Red Tree” (1908) ve sağ alt köşede “Composition with red, yellow, blue, and black” (1923) çalışmalarına kadar gerçekleştirdiği tablolar arasında, Mondrian'ın kavramsal soyutlamada yaklaşımındaki değişimi takip etmek mümkün hale gelmektedir.



Şekil 3.1 : Sanatçı soyutlamasına örnek olarak Piet Mondrian’ın çalışmaları (Url- 9).

Ancak hem kavramsal hem prosedürel, her iki soyutlama yaklaşımı için de doğru genellemeler yapabilmek önemlidir (Wing, 2008; Arnheim, 2004). Bu bakımdan, sanat ve tasarım disiplinleri için soyutlama, algı ile yakından ilişkili ve kavramsal boyutta iken; bilgisayar bilimleri ve diğer mühendislik alanları için bu kavramların daha çok prosedürel düzeyde kullanıldığını söylemek mümkündür. Prosedürel soyutlama, kavramsal soyutlamaya karşın daha net ve bilişim araçlarıyla mümkün olan en basit şekilde ifade edilebilir düzeydedir.

Diğer yandan, başarılı bir soyutlama için, ayrıştırma ve örüntü tanıma becerilerinin kullanımıyla ortaya çıkan genellemelere yer verir. Arnheim’a göre (2004), “gerçek genelleme, bir bilim insanının kavramlarını, bir sanatçının ise imajlarını mükemmelleştirmesinin bir yoludur ve bu kesinlikle mekanik olmayan bir prosedürdür.” Bununla birlikte, görsel bir düşünür için genelleme, rastgele veya sonsuz sayıda örnek toplama meselesi değildir. Buna bağlı olarak “görsel düşünmenin gerçek gücü, birçok yönden örüntü yakalamayla ortaya çıkar ve örüntü seçimi, izleyici tarafından işlenecek görsel sorgular için önemlidir” (Ware, 2008; s:174).

Kısaca, soyutlama altında örüntü tanıma, ayrıştırma ve genellemeler için pratik yapmak, eğitimli bir göz sahibi olmaya yardımcıdır (Şekil 3.2). Bu tür algısal görevler, temel beceriler üzerine inşa edildiğinde daha kolay ve hızlı gelişim gösterir (Ware, 2008; s:172).



Şekil 3.2 : Görsel düşünürlerin örüntü tanımayla ihtiyaç duyduğu eylemler (Ware, 2008).

Hiç şüphesiz ki bu kapsamda mimarlık eğitiminde görsel düşünmenin hesaplama ile ilişkilendirildiği en yaygın model, George Stiny ve James Gips (1971) tarafından geliştirilen Biçim Gramerleri (Shape Grammars) olmuştur. Bu model, “semboller, kelimeler, sayılar veya diğer soyut biçimler yerine doğrudan iki veya üç boyuttaki şekillerle hesaplama” yapmayı önermektedir (Knight, 1999; Stiny 2006).

Stiny’nin ifadesine göre (2015) biçim gramerleri, görsel hesaplama üzerinden yapma eylemi ilişkilendirmesi ve bu eylem için yeni yönelimleri desteklemesi nedeniyle diğer hesaplamalı tasarım yaklaşımlarına kıyasla farklılık gösterir (s:26). Tasarım tanımlamak ve üretmek için kullanılan bu kural tabanlı sistem (Stiny ve Gips, 1971) temel tasarım eğitiminde analog uygulamalar olarak sıkça kullanılmaktadır (Smitswick, 2015). Ancak biçim gramerleri ile elde deneyim uygulamaları, daha çok soyutlamanın kavramsal yönüyle ve yaratıcı düşünme ile ilişkilendirilmektedir. Bu çerçevede, farklı görme biçimlerini desteklemek ve pekiştirmek açısından biçim gramerleri oldukça değerli olsa da; bu modelin kullanıldığı analog uygulamalarda davranış örüntülerinin göz ardı edildiğini söylemek mümkündür. Nitekim, bir tasarımcının örüntüye gösterdiği tepki de bir çeşit örüntüdür ve bu tepki örüntüsü algıyı eyleme bağlayan becerilerin özüdür (Ware, 2010; ss:173-4). Dolayısıyla, bu tür uygulamalarda, soyutlamayı prosedürel düzeyde tanımlamak ve anlamlandırmak oldukça zordur.

Bu durumu basit bir geometri uygulaması ile örneklendirmek mümkündür. Şekil 3.3’te yer alan geometriyi bir üçgeni orantılı olarak küçülterek tekrarlamayla ortaya çıkarmak mümkünken, aynı zamanda, küçük bir üçgenin büyütülerek tekrarlanması ile üretmek de mümkündür.



Şekil 3.3 : Basit geometrik şekillerden oluşan örüntüler.

Bu iki yöntemle aynı sonuca ulaşmak mümkün olsa da gidiş yolları, nesnenin algılanma biçimi görsel hesaplamanın kurgusunu önemli şekilde değiştirecektir. Bu nedenle biçim gramerleri temel tasarım eğitimi içinde oldukça elverişli bir model olsa da dijital araçlar ve etkileşimli tasarım ortamlarının kullanımı söz konusu olduğunda, örüntü tanıma ve ayrıştırma becerileri, davranış örüntüleri yönünden aynı elverişliliğe sahip değildir. Soyutlamada oluşturulan (görme ve davranış biçimi olarak) farklı türdeki örüntülerin birbirleriyle ilişkilendirilmesi önemlidir.

3.2 Tasarımcı Bakış Açısıyla Problem Formüle Etme: Algoritmik Düşünme

Algoritmalar bir şeyi uygun biçimde yapmak için açıklamada kullanılan talimatlardır (Reas ve McWilliams, 2010; s:13); ve genellikle çözümü bilgisayarda işlenebilecek şekilde oluşturmada kullanılırlar (Combéfis, Van Den Schrieck ve Nootens, 2013; s:4). Bu nedenle, geleneksel düşünme biçimimize kıyasla farklılık gösterir. Her ne kadar bu düşünme biçimi bir tür makine becerisi olarak atfedilmiş olsa da algoritma oluşturmak, insan zihni için de oldukça elverişli bir düşünme biçimidir ve örgü örmeden yön tarifine kadar pek çok gündelik olayda yer alır.

Gerald Futschek'e göre (2006) “algoritmik düşünme, algoritma oluşturmak ve algoritmaları anlamak için ihtiyaç duyulan bir beceri havuzudur” ve bu havuz,

- Verilen problemleri analiz etme;
- Bir problemi net olarak ifade etme;
- Verilen probleme karşılık en basit eylemle harekete geçme;
- Verilen problemler için basit eylemlerle doğru algoritmayı oluşturma;
- Problem için tüm olası ve özel senaryolar hakkında düşünme ve buna bağlı olarak algoritmayı iyileştirme becerilerini içerir (s:160).

Bilgisayar Bilimlerinde algoritmik düşünme için sunulan öğrenim deneyimleri genellikle C+, Java, Python gibi yaygın programlama dilleri ile kodlama (code) ve sözde kodlama (pseudocode) pratiklerini içermektedir. Ancak, bu tür programlama dilleri ifade yönünden, alan dışı ve acemi bireyler için fazla soyut olabilir (Futschek ve Moschitz, 2010; s:3). Temel eğitimi bedensel deneyim ile kazanan tasarımcıya algoritmik düşünme becerisi kazandırmak için programlama dilinin öğretimi ile başlamak oldukça zorlayıcıdır. Görsel bir düşünür olarak tasarımcının algoritmik düşünme deneyimini görsel uyarıcılara yer veren uygulamalar çerçevesinde kazanması oldukça önemlidir.

Ayrıca algoritmik düşünme, doğru uygulama yöntemi ve araçlarla kazandırıldığında bir tasarımcı için oldukça işlevseldir. Kostas Terzidis'in ifade ettiği üzere (2006) "Algoritmaların artikülasyonu ile tasarımın benzerlikleri dikkat çekicidir. Böylesi bir ilişkilendirme, tasarım için faydalı prosedürleri geliştirmeye yardımcı olurken, tasarım hakkında zihinde oluşan bazı ipuçlarını ortaya çıkarabilir (s:20)" Terzidis'in yaklaşımı doğrultusunda bu düşünme biçimini tasarım ile ilişkilendirdiğimizde ise algoritmik düşünmenin aslında bir tasarımcının mantıksal analiz sürecinde sıkça başvurduğu bir düşünme biçimi olduğu görülmektedir. Çünkü "tasarım süreci hem mantıksal hem de yaratıcı düşünceyi kapsayan iki veya daha fazla eş zamanlı düşünce raylarına ihtiyaç duyar ve veri toplama, problem tanımı ve çözüm değerlendirmesi gibi analitik faaliyetlerle tasarım bağlamının yeniden çerçevelenmesi olabildiğince açık ve objektif olmalıdır" (Tovey, 2015; s:54).

Bu bağlamda tasarımcının algısal yaklaşımı çerçevesinde oluşturduğu algoritmik düşüncenin tamamen nesnel manipülasyonlardan ibaret olmadığını söylemek mümkündür. Dolayısıyla, bir tasarımcının hem tanımlı problemler karşısında hem de süreç içinde olası yeni problemlerle başa çıkabilecek donanımına sahip olması gerekir. Tasarımcı karşılaştığı farklı problemleri çözmek ya da formüle etmek için algoritmik düşünmeye ihtiyaç duyar.

Ve bu ihtiyaç, tasarım ile algoritmik düşünme arasındaki benzerlikleri ortaya çıkarır. Kısaca algoritmik düşünme tasarımcı bakış açısıyla problem çözmedir ve görsel bir düşünür için algoritma oluşturmak, algısal ve sezgisel yaklaşımlara öncelik tanımayı gerektirir. Bir tasarımcının algoritmik düşünme için ihtiyaç duyduğu öğrenim deneyiminde algısal yaklaşımlara yer vererek esneklik payı bulundurmak önemlidir.

Bu gerekçeyle Derek Ham (2015) Futchek'in algoritmik düşünme tanımını, tasarımcı bakış açısını dikkate alarak şu üç yeni madde ile zenginleştirir (ss:90-91):

- Sürekli değişen bir problemi analiz edebilme,
- Temel hareketlere bağlı olarak yeni hareketler bulma ve
- Verilen problemdeki değişikliğe yanıt verebilecek şekilde esnek bir algoritma oluşturma.

Sonuç olarak HD'nin, temel bilişsel değerlerindeki olası kavramsal farklılıkların mimarlık eğitim müfredatında belirleyici oldukları göz ardı edilmemelidir. Bu nüans, bir tasarımcının hesaplamalı düşünme sürecinde kullandığı soyutlama becerisinin, kavramsal ve prosedürel olmak üzere, farklı düzeylerde değerlendirilmesi ve geliştirilmesi gerektiğinin önemli bir göstergesidir. Bu durum, aynı zamanda mimarlık eğitiminde HD'nin öğretim kültürü çerçevesinde, özelleştirilebilir/bireyselleştirilebilir öğrenim deneyimleri sunabilmenin pedagojik yönden önemini ortaya koymaktadır.

Bu kapsamda, bir sonraki bölümde, tasarımcının kavramsal ve prosedürel düzeyde soyutlama becerisini tasarım bilişim araçlarının kullanımı çerçevesinde değerlendirilmektedir. Bunun için, tasarımcıların geleneksel CAD programlarını kullanım süreçleri sistematik bir analiz çerçevesinde ele alınmış; tasarımcıların soyutlama becerilerini kavramsal ve prosedürel düzeyde ele alan bir değerlendirme yöntemi geliştirilmiştir. Bu yöntem, farklı disiplinlerden katılımcıların yer aldığı uygulamalı bir deneyde kullanılmıştır.

4. TASARIMCI BAKIŞ AÇISI İLE HESAPLAMALI DÜŞÜNMEDE SOYUTLAMAYA DENEYSEL BİR YAKLAŞIM

Önceki bölümlerde vurgulandığı üzere, mimarlık eğitim alanında, tasarım öğrencilerinin HD sürecine bilişsel yönüyle değer veren çalışmaların oldukça az olduğu görülmektedir. Yine bu çalışmalarda, soyutlama ve soyut düşünmenin ağırlıklı olarak kavramsal düzeyde ele alındığını söylemek mümkündür. Bu çalışmada ise farklı bir strateji benimsenmiş; mimarlık öğrencilerinin hesaplamalı düşünme süreçlerinin sistematik analizi ile, kavramsal ve prosedürel düzeyde sahip oldukları soyutlama becerilerini değerlendirmek amaçlanmıştır.

Gerçekleştirilen çalışma için, seçilen katılımcı gruplar arasında tasarım uzmanlıkları ve program kullanım becerileri yönünden düzey farklılıklarının olması tercih edilmiştir. Bu kapsamda, temel tasarım eğitimini tamamlamış; proje dersleri ile tasarım deneyimi ve problem çözme deneyimi kazanmış mimarlık öğrencilerinin; tasarım alanı dışından olmak koşuluyla farklı disiplinlerden lisans düzeyinde bir kontrol grubuyla karşılaştırılması uygun görülmüştür.

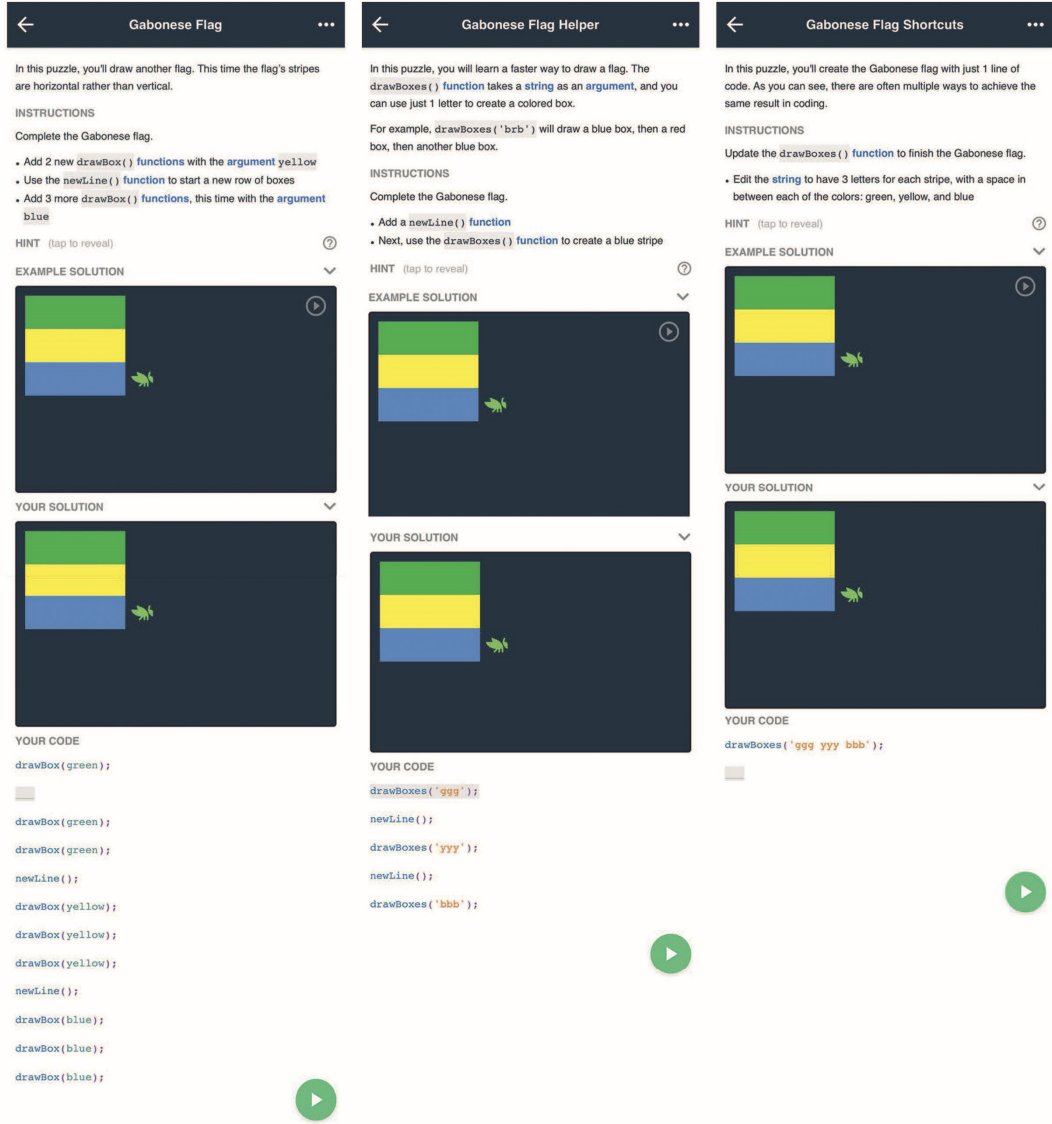
Çalışma için, mimarlık öğrencilerinin görsel imgeler üzerinden kavramsal düzeyde soyutlama yapmalarını hem de hesaplamalı düşünme çerçevesinde prosedürel düzeyde soyutlama yapmalarını gerektiren bir uygulamaya ihtiyaç duyulmuş; bu kapsamda bir araştırma kurgulanmıştır. Bunun için, literatürden ‘Yeni Başlayanlar için HD Eğitimi’ kapsamında sunulan uygulama örnekleri ele alınmış; bu örneklerde, yapılan ‘soyutlama’ tanımları ve ‘soyutlama becerisi’nin değerlendirilmesinde kullanılan yöntem ve stratejiler incelenmiştir. Ardından, incelenen yöntemlerin görsel bir düşünür için CAD modelleme ortamına nasıl uyarlanabileceğine dair bir araştırma yapılmış; bu doğrultuda mimarlık öğrencileriyle bir dizi pilot çalışma gerçekleştirilmiştir. Pilot çalışmalar, deney için uygulama ortamı ve araçlarını seçmeye; uygulama içeriğini tanımlamaya ve değerlendirme yöntemini geliştirmeye yardımcı olmuştur.

4.1 Araştırmanın Kurgusu

4.1.1 Yeni başlayanlar için hesaplamalı düşünmede prosedürel soyutlama

Hesaplamalı Düşünme sürecinde prosedürel düzeyde soyutlama yapabilmek, sürecin etkinliği ve verimliliği açısından kritik öneme sahiptir. Çünkü programlama esnasında prosedürel soyutlama yapmak hem gereksizce tekrarlayan kodlamalardan kaçınmayı sağlar hem de programın kurgusunu sağlamlaştırır ve kontrolünü kolaylaştırır. Bu nedenle, yeni başlayanlar için HD pratiğini programlama yoluyla kazandıran ve bireyin bilişsel becerilerini nicel yönden değerlendiren eğitim uygulamalarının çoğunda, prosedürel düzeyde soyutlama becerisi kazandırmak, öğrenim hedeflerinin içindedir. Bu bağlamda, prosedürel soyutlamanın tanımı ve ele alınış biçimine örnek teşkil etmesi için, ‘betik programlama’ (script programming) ve ‘bloklarla programlama’ (block-based programming) çerçevesinde geliştirilen iki farklı eğitim uygulamasına yer verilmiştir.

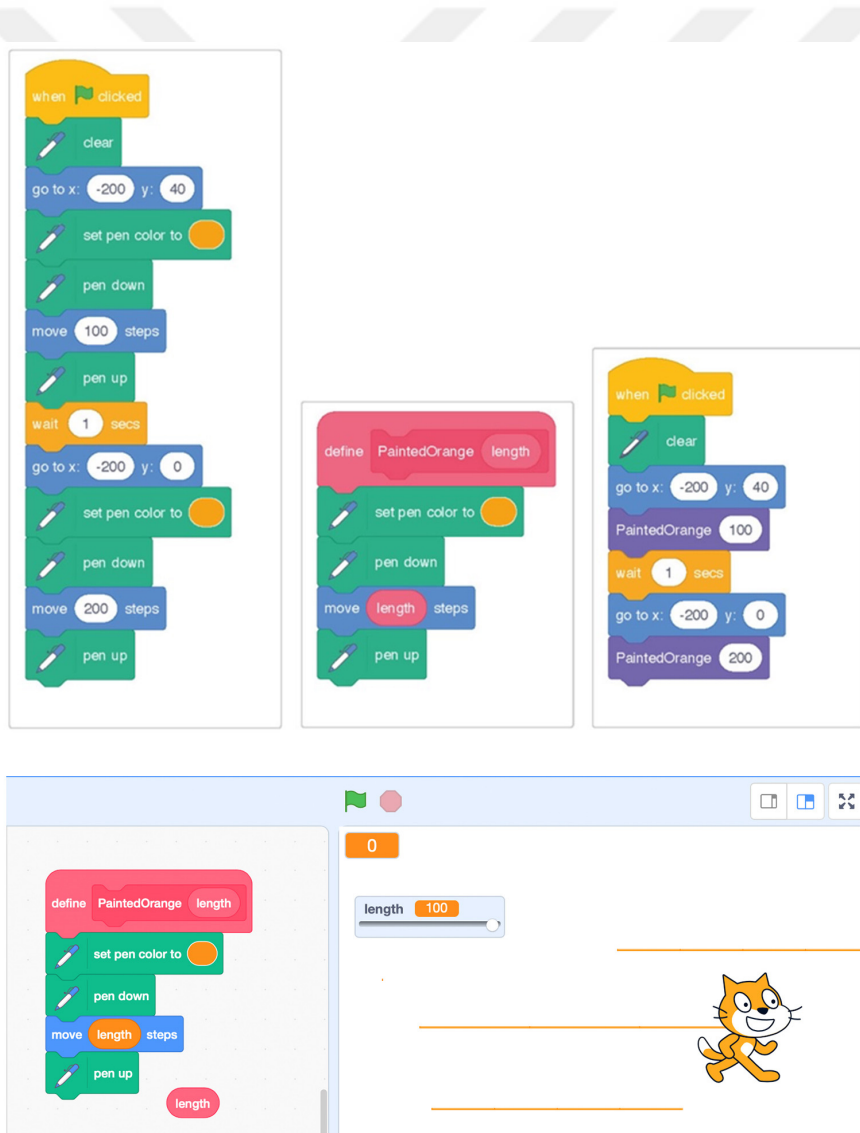
İlk olarak, betik programlama eğitimi için Google tarafından geliştirilen Grasshopper ve bu uygulamada sunulan ‘Bayrak Kodlama’ egzersizlerini örnek göstermek mümkündür. Google Grasshopper, Java Script’in betik dili üzerinden yeni başlayanlara kodlama eğitimi sunmak üzere geliştirdiği bir mobil cihaz uygulamasıdır. Bu uygulamada kullanıcı, görsel örnekler ile, aşamalı olarak hesaplamalı düşünmeye ve programlama pratiği yapmaya teşvik edilmektedir. Uygulama içinde farklı aşamalarda sunulan Bayrak Kodlama egzersizlerinde ise kullanıcı, prosedürel düzeyde soyutlama ile tanıştırılmaktadır. Şekil 4.1’de görüldüğü üzere, bayrak çizmek olarak tanımlanan bu uygulama problemi için, sırası ile üç farklı çözüm yöntemi kullanıcıya sunulmaktadır. Bu kapsamda kullanıcı, sırayla fonksiyon tanımlamak, dizi oluşturmak ve değişken tanımlamak üzere, programlamada kullandığı değişken ve fonksiyon benzerliklerine bağlı olarak genellemeler yapmayı, prosedürel düzeyde soyutlama yapmayı deneyimlemektedir.



Şekil 4.1 : Bayrak Kodlama egzersizlerinden ekran alıntıları, Google Grasshopper Mobil Uygulaması, 2019.

Benzer şekilde, yeni başlayanlar için HD kapsamında kullanılan bir başka görsel programlama uygulaması Scratch ve Scratch'a özgü geliştirilmiş olan ölçme ve değerlendirme uygulaması Dr. Scratch'ı da örnek olarak göstermek mümkündür. MIT Media Lab tarafından geliştirilen, blok-tabanlı bir programlama dili olan Scratch, özellikle ilköğretim düzeyindeki kullanıcılara görsel programlama ile beceri kazandırmada kullanılmakta; bu kapsamda HD'ye yer veren eğitim araştırmalarında sıkça karşımıza çıkmaktadır (örnek: Topallı ve Çağıltay, 2018; Romero, Lepage ve Lille, 2017).

Dr.Scratch ise Scratch projelerinin içeriğini Soyutlama, Parallellleştirme, Mantıksal Düşünme, Senkronizasyon, Akış Kontrolü, Kullanıcı etkileşimi ve Veri Gösterimi kavramları açısından analiz etmek ve puanlamak üzere eğitimciler için geliştirilmiş bir web aracıdır. Bu araç, soyutlama becerisini prosedürel yönden değerlendirmek için, Scratch ile üretilen programlarda kodlama örüntülerini analiz eder (Dr.Scratch, 2019). Verilen görsel bir problem karşısında oluşturulan çözüm yollarının/programın içinde gereksizce tekrar eden prosedürleri tespit ederek puanlama yapar. Şekil 4.2’de görüldüğü üzere, Scratch’ta ‘iki turuncu çizgi çizmek’ olarak tanımlanan bir problem için sunulan farklı çözüm önerileri yer almaktadır. Sunulan bu önerilerin içeriğinde, gereksiz tekrarlardan kaçınarak en az blokla tanımlanan çözüm önerisi, Dr.Scratch’ta daha başarılı bir çözüm olarak değerlendirilmektedir.



Şekil 4.2 : Scratch uygulamasında bir çizim problemi ve üretilen farklı çözüm önerileri (Url-10).

Her iki uygulamada da prosedürel soyutlama, verilen problemler için sunulan çözüm önerilerinde değişkenler ve fonksiyonlar arasındaki benzerliklere bağlı olarak yapılan gruplandırmalar çerçevesinde dikkate alınmıştır. Bu kapsamda, her iki uygulamaya bakarak, HD’de prosedürel soyutlama becerisinin, tanımlı bir problem karşısında oluşturulan çözüm önerisinde yer alan değişkenlerin ve fonksiyonların arasındaki benzerliklerin nitelikli gruplandırılması ile yakından ilişkili olduğunu söylemek mümkündür. Dolayısıyla, soyutlama becerisini prosedürel yönüyle değerlendirmek için, bireyin, tanımlı bir problem karşısında geliştirdiği çözüm önerisi bağlamında, kullanmış olduğu veri ve gerçekleştirdiği hesaplamalı işlemler çerçevesinde oluşturduğu genellemeler dikkate alınmalıdır.

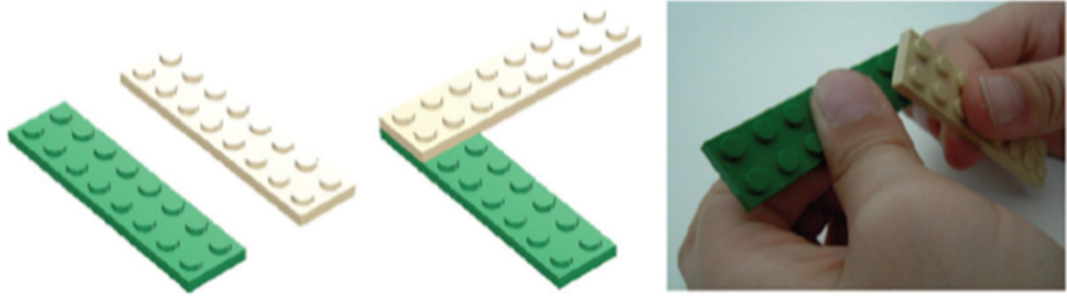
Bu nedenle, araştırma hedefi doğrultusunda, mimarlık öğrencilerinin soyutlama becerilerini hem prosedürel hem de kavramsal yönleriyle değerlendirebilmek için, deneyin tasarım bilişim araçlarının kullanımı çerçevesinde gerçekleştirilmesi uygun görülmüştür. Bu doğrultuda, mimarlık eğitiminde yaygın olarak kullanılan modelleme programları ve araçlarının kullanımı çerçevesinde bir deney uygulaması için araştırma yapılmıştır.

4.1.2 Geleneksel CAD araçları ile hesaplamalı düşünme

Tasarım bilişim araçlarının kullanımı çerçevesinde mimarlık öğrencilerinin hesaplamalı düşünme süreci, ağırlıklı olarak ‘parametrik düşünme’ ‘algoritmik düşünme’ gibi özelleşen konu başlıkları altında karşımıza çıkar (örnek: Çolakoğlu ve Yazar, 2007; Oxman, 2017; Yazıcı, 2019). Bu çalışmalarda, hesaplamalı düşünmenin öğretim kültüründe yaygın olarak kullanılan Grasshopper, Generative Components, Dynamo gibi görsel programlama araçları kullanılmakta; ancak öğrencilerin HD süreci, karmaşık tasarım problemleri çerçevesinde ve nitel yaklaşımlarla değerlendirmektedir.

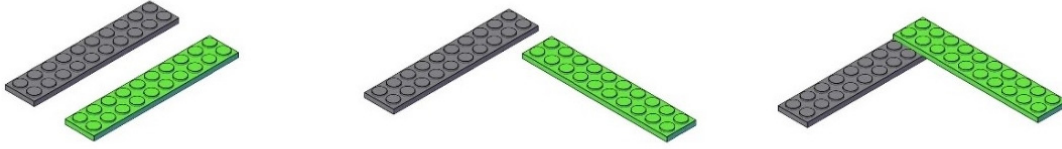
Diğer yandan tasarımcının HD sürecinin sistematik analizi çerçevesinde ise hem bu tür görsel programlama araçlarını kullanmanın hem de bu araçların kullanımına karmaşık tasarım problemleri çerçevesinde yer vermenin elverişsiz olduğunu söylemek mümkündür.

Çünkü bu tür programlar çalışma prensipleri gereği iki türlü bilgi kullanımını gerektirir: Bunlardan ilki olan komut bilgisi, algoritmalar veya araçların yazılım içinde kullanım prosedürleri iken; ikincisi olan stratejik bilgi, belirli bir görevi gerçekleştirmek üzere oluşturulan alternatif yöntemleri ve bu yöntemlerde kullanılan işlemlerin hangi surette gerçekleştirilebileceği ile ilgilidir (Chester, 2007). Nitekim, CAD programları ile gerçekleştirilen eylemler de geleneksel ortamda gerçekleştirilen eylemlerden farklı olarak sıralı ve kurallı biçimde düşünmeyi gerektirir. Bu duruma örnek olarak, iki Lego© parçasını birleştirme işlemini bilgisayar ortamında gerçekleştirmenin, elde gerçekleştirmekten farklı şekilde düşünmeyi gerektirdiği gösterilebilir (Şekil 4.4). Bir başka ifade ile, geleneksel CAD programlarında çalışırken kullanılan taşıma (move), seçme (select), döndürme (rotate) gibi eylemler, her ne kadar analog ortamda gerçekleştirilebilir olursa olsun, bu eylemleri CAD arayüzünde gerçekleştirebilmek, çoğu zaman talimatlar tanımlamak ve bu talimatları sıralı ve kurallı biçimde uygulamakla mümkündür.



Şekil 4.4 : “İki Lego© parçasını gerçek hayatta ve dijital ortamda birleştirmek, uzamsal ilişkileri farklı detaylarda planlamayı gerektirir.” (Erhan, Youssef ve Berry, 2012; s:384).

Geleneksel CAD programlarında, kullanıcının arayüz etkileşimi sırasında otomatik olarak kayıt altına alınan talimatların betikleri, İnsan-Bilgisayar Etkileşimi (HCI) alanında yapılan araştırmalarda veri olarak kullanılmaktadır (örnek: Bhavnani ve diğ., 1999; Sadeghi v.d., 2016). Geleneksel CAD programlarında bu betikler, kullanıcının arayüz etkileşimindeki izlenen prosedürel süreci yansıtırken, kullanıcıların yaptıkları seçimleri takip etmeye, görme ve davranış örüntülerini yakalamaya yardımcı olmaktadır (Şekil 4.5).



```
Command: ROTATE  
Current positive angle in UCS: ANGDIR=counterclockwise ANGBASE=0  
1 found  
Specify base point:  
Specify rotation angle or [Copy/Reference] <270>:  
>>Enter new value for ORTHOMODE <0>:  
Resuming ROTATE command.  
Specify rotation angle or [Copy/Reference] <270>:  
Command: Specify opposite corner or [Fence/WPolygon/CPolygon]:  
Command:  
Command: M  
MOVE: 1 found  
Specify base point or [Displacement] <Displacement>:  
Specify second point or <use first point as displacement>:
```

Şekil 4.5 : AutoCAD ortamında seçilen bir LEGO® parçasını diğeri ile birleştirmede yer alan talimat dizisi.

Bu kapsamda, deney uygulamalarında, katılımcıların katı nesne modelleme esnasında seçtikleri geometriyi ve geometri uyguladıkları eylemleri takip edebilmek ve arayüz etkileşimi sırasında gereksizce tekrar ettikleri eylemleri tespit edebilmek için, geleneksel CAD programı içinde otomatik olarak kayıt altına alınan betiklerin kullanımına karar verilmiştir. Sunulacak deney uygulamasında kullanılabilecek arayüzün tanımlanması ve uygulama içeriğinin geliştirilmesi için de bir dizi pilot çalışma gerçekleştirilmiştir.

4.2 Pilot Çalışmalar

Gerçekleştirilen literatür araştırması neticesinde, geleneksel CAD programları ile çalışan tasarımcıyı daha yakından tanımak ve bu çerçevede deney uygulamasının içeriğini tanımlamak amacıyla üç bölümden oluşan bir dizi pilot çalışma kurgulanmıştır. 2B ve 3B geometri modelleme uygulamalarından oluşan bu çalışmalar, Şekil Yapma Çalıştayı (Shape Making Workshop) altında; üçüncü ve dördüncü yıl mimarlık öğrencileri ile gerçekleştirilmiştir. Ortalama 90 dakika süren uygulamalardan oluşan çalıştaya, öğrenciler gönüllü olarak katılmıştır. Bu nedenle, öğrencilerin kimlikleri gönüllülük esasına binaen saklı tutulmuş; yapılan gözlem ile elde edilen veri içeriği, tezde anonim olarak paylaşılmıştır.

Çalıştay içeriği Çizelge 4.1’de özetlenmiştir.

Çizelge 4.1 : Pilot çalışmaların yer aldığı Şekil Yapma Çalıştayı'na ait içerik çizelgesi.

Süre	Uygulama Adı	Uygulama İçeriği	Uygulama Hedefi	Toplanan Veri Türü
Hafta 1	SOMA OYUN BLOKLARI			
	Öğretici Başlangıç	3B katın nesne modelleme ve tasarım	Geleneksel CAD uygulama ortamının kararlaştırılması	Video Ekran Kayıtları
	Bölüm1: Blok Modelleme Bölüm 2: Kule tasarımı oyunu	TeamViewer'da İşbirlikçi Tasarım Uygulaması		Uygulama Dosyası
Hafta 2	ŞEKİL YAPMA			
	Öğretici Başlangıç	2B şekiller üzerinden hesaplamalı tasarım uygulaması	AutoCAD uygulama ortamında çizgisel ve katı nesnelerin oluşturduğu algısal farklılıkların tasarım sürecine etkisi	Video Ekran Kayıtları Uygulama Dosyası
	Tek Bölüm: Şekil Türetme			Komut Geçmişi Kaydı
Hafta 3	YÜZEYLER			
	Öğretici Başlangıç	Mirror3D için uygulama alıştırmaları	3B şekiller üzerinden Görsel hesaplama	Video Ekran Kayıtları
	Bölüm1: Alıştırma	3B Geometrilere Parçala/Birleştir tasarım ve modelleme Uygulaması	Komut kullanımındaki davranış örüntüleri	Uygulama Dosyası
	Bölüm 2: Modelleme Bölüm 3: Yüzey Oyunu	TeamViewer'da İşbirlikçi Tasarım Uygulaması	Kavramsal ve Prosedürel Soyutlamaların Saptanması	Komut Geçmişi Kaydı

4.3 Katılımcılar

Çalıştay katılımcıları, mesleki eğitimi çerçevesinde problem çözme deneyimi kazanmış ve geleneksel CAD programlarını kullanmada deneyim sahibi öğrencilerden oluşmaktadır. Bu öğrencilerin, program kullanım becerilerini öz-yönlendirmeli olarak (self-regulated learning) kazanmış olmalarına dikkat edilmiştir.

Kees Dorst'un (2015) tasarım uzmanlığı için nitelendirdiği tabloya göre, çalıştay katılımcılarını mesleki anlamda tek başına problem çözme deneyimi kazanmış 'acemi tasarımcılar' olarak nitelendirmek mümkündür. Bu kapsamda, İTÜ Mimarlık Fakültesi'nde altı ve sekizinci yarıyıldan, 16 mimarlık öğrencisinin gönüllü katılımıyla gerçekleştirilmiştir. Çalıştay bilgisayar laboratuvarında gerçekleştirilmiş ve tüm uygulamalarda öğrenciler kendi bilgisayarlarını kullanmıştır (Şekil 4.6). Çalışmalara katılan öğrencilerin yetkin oldukları bilgisayar programları ve öğrenim deneyimlerine ait grafik dağılım, Şekil 4.7 ve Şekil 4.8'de sunulmuştur.



Şekil 4.6 : Bilgisayar laboratuvarında çalıştay uygulama ortamından bir görüntü.



Şekil 4.7 : Pilot çalışmalarda yer alan katılımcıların öz-yönlendirme ile öğrendikleri bilgisayar programlarının grafik dağılımı.



Şekil 4.8 : Pilot çalışmalarda yer alan katılımcıların iki boyutlu uygulamalarda kullandıkları bilgisayar programlarının grafik dağılımı.

Çalıştay sonunda, öğrencilere yönlendirilen uygulama değerlendirme anketi EK A’da yer almaktadır. Bu ankette, öğrencilere gerçekleştirdikleri uygulamaları değerlendirmeleri için sorular yöneltilmiştir. Ayrıca öğrencilerden, uygulamalarda kullanmış oldukları soyutlama, örüntü tanıma, ayrıştırma ve algoritmik düşünme becerileri için öz değerlendirmede bulunmaları istenmiştir. Yapılan değerlendirme için likert ölçeği kullanılmıştır.

4.3.1 Uygulamaların içeriği ve görevlerin tanımı

Önceki bölümde bahsedildiği üzere, yapılacak deneyin farklı düzeyde tasarım uzmanlığına sahip katılımcılarla gerçekleştirilmesi amaçlandığından; bu aşamada gerçekleştirilecek olan tüm uygulamaların arayüz etkileşimi ve araç kullanımı konusunda zorluk oluşturmayacak nitelikte olmasına özen gösterilmiştir. Dolayısıyla, katılımcılara hem oyun bloklarıyla oynamak kadar zengin içerikte hem de sürükle ve bırak (drag and drop) deneyimi sunacak kolaylıkta bir uygulama ortamı ve deney uygulaması tasarlamak amaçlanmıştır. Çalıştay için hazırlanan tüm uygulamalarda oyunlaştırma yöntemi benimsenmiştir. Bu kapsamda, her uygulama içinde öğrencilere çeşitli görevler tanımlanmış ve öğrencilerden, tek başlarına veya iki kişilik grup oluşturarak bu görevleri tamamlamaları istenmiştir.

Görev olarak öğrencilere, geleneksel CAD programları ile katı nesne modelleme çerçevesinde, özel olarak hazırlanan 2B/3B Şekil Yapma uygulamaları sunulmuş; sunulan görevler ile mimarlık öğrencileri, soyutlama, örüntü tanıma, ayırıştırma ve algoritma oluşturma becerilerini uzamsal çerçevede kullanmaları için teşvik edilmiştir.

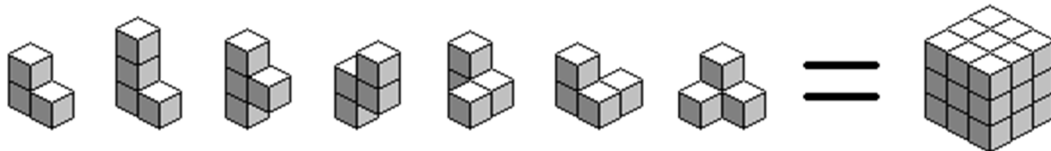
Uygulamalarda verilen görevlerin ve seçilen modelleme programının, bireyin arayüz etkileşimi süresince ortaya çıkartacağı olası görme ve davranış örüntülerini yansıtabilecek nitelikte olmasına dikkat edilmiştir. Değerlendirmede, yürütücü gözleminde alınan notların yanı sıra, öğrencilerin bilgisayarlarıyla aldıkları video ekran kayıtları ve uygulama dosyalarından yararlanılmıştır.

Çalıştayda öğrenciler sunulan uygulamaların detaylı içeriği ve uygulama süresince yapılan gözlemlerden elde edilen çıkarımlara sırayla aşağıda açıklanmıştır.

4.3.1.1 Uygulama 1: SOMA oyun blokları

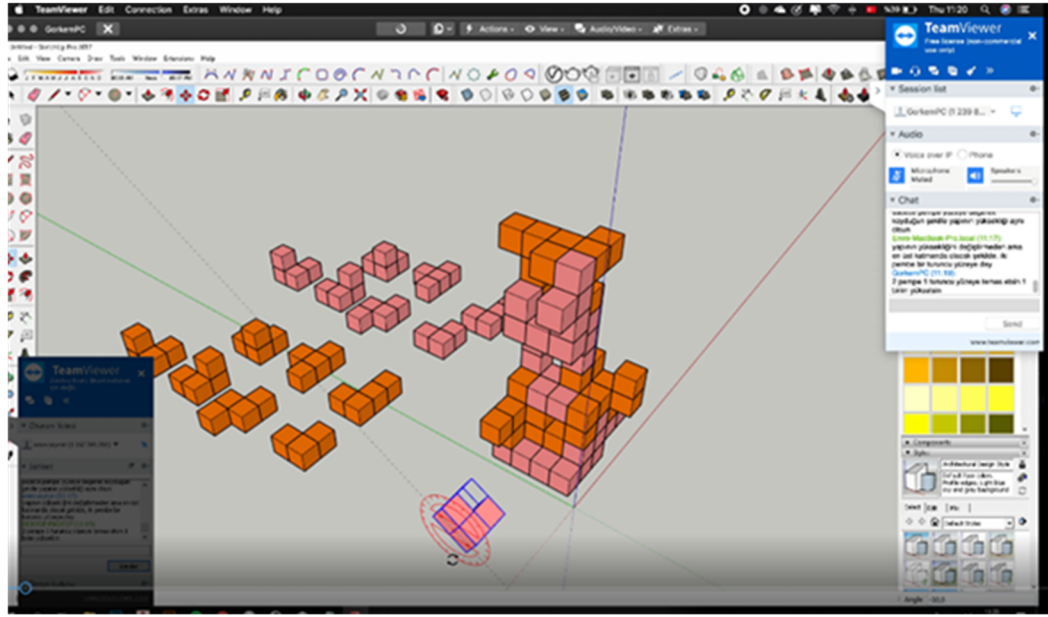
Çalıştayın ilk uygulamasında, farklı CAD modelleme programlarını tercih eden öğrencilerin modelleme sürecinde kullandıkları dijital araçları ele alış biçimleri ve arayüz kullanım tercihleri dikkate alınmıştır. Yapılan gözlem sonucunda, hangi programın deney uygulamalarının geliştirilmesi ve sistematik analiz için elverişli olduğuna karar verilmiştir.

Verilen ilk görevde, öğrencilerden ikili gruplar oluşturmaları ve TeamViewer'ı (Ekran Paylaşım Yazılımı) kullanarak, tercih ettikleri bir geleneksel CAD modelleme programında ortak bir çalışma ortamı tanımlamaları istenmiştir. Ardından her bir grup üyesinin, tercih ettiği yöntemle Şekil 4.9'da tanıtılan SOMA Oyun Bloklarını modellemesi istenmiştir. Bu süreçte, öğrencilerin katı nesne modelleme için tercih ettikleri program, şekil üretme yöntemleri ve kullandıkları komutlar dikkate alınmıştır.



Şekil 4.9 : SOMA Oyun blokları, (Url-11).

Uygulamanın ikinci bölümü için bir Kule Oyunu kurgulanmıştır (Şekil 4.10). Bu oyun için, öğrencilere birtakım talimatlar ve sınırlamalar tanımlanmış; bu sınırlamalar çerçevesinde öğrencilerden, ilk bölümde modelledikleri bloklarla, (TeamViewer ekranı üzerinden) işbirlikçi ortamda çalışarak, bir kule tasarımları istenmiştir. Oyun süresince öğrencilerden, grup arkadaşlarıyla konuşmadan iletişim kurmaları, birbirlerine yazılı talimatlar göndererek (blok seçimi yaparak ve konum tanımlayarak) çalışmaları istenmiştir. Bu süreçte, öğrencilerin yazılı olarak talimat vermede ve grup arkadaşlarından aldıkları yazılı talimatları gerçekleştirmede ne kadar başarılı olduklarına dikkat edilmiştir. Ancak, öğrencilerin kullandıkları modeller için yazılı olarak yaptıkları şekil ve konum tanımlamalarında, ağırlıklı olarak doğal dil kullanımını tercih ettiklerini ve bu nedenle grup üyeleriyle sıkça anlaşmazlıklar yaşadıklarını belirtmek mümkündür.



Şekil 4.10 : Blok Modelleme ve Kule Oyunu sürecinden ekran alıntısı.

Blok Modelleme ve Kule Oyunu için öğrencilere verilen talimatlar ve sınırlamalar sırayla şu şekildedir:

- Uygulamada birlikte çalışmak üzere bir grup arkadaşı seçin ve TeamViewer ile bu arkadaşınızın bilgisayarına bağlanın;
- 3B katı nesne modellemek için zorluk çekmediğiniz geleneksel bir CAD programı seçin ve bu programda çalışmak üzere ortak bir çalışma alanı tanımlayın;

- Uygulama için gösterilen size verilen SOMA oyun bloklarının her bir parçasını birim küplerden oluşmak kaydıyla modelleyin ve bu parçaları, siz ve grup arkadaşınız olmak üzere, kendi adlarınız altında birer renkli katmanla tanımlayın;
- Farklı bir katmanla, bir kule tasarlamak üzere kullanacağınız 5x5 birim boyutlarında bir uygulama alanı (taban) tanımlayın;
- Dönüşümlü olarak, grup arkadaşınıza bir parça ve bir konum talimatı vererek (TeamViewer'deki sohbet çubuğunu kullanarak) bu alanda bir kule tasarlayın;
- Kuleniz toplamda 25 birim blok yüksekliğine ulaşana dek dönüşümlü çalışmaya devam edin;
- Verilen yüksekliğe ilk ulaşan oyunu kazanır.

Uygulamada katılımcıların Google SketchUp, Autodesk AutoCAD ve 3DsMax modelleme programlarını tercih etikleri görülmüş; ancak arayüz kullanım kolaylığı ve sistematik analiz için deneyde kullanılacak programın AutoCAD olmasına karar verilmiştir. Katılımcıların video ekran kayıtları incelendiğinde, AutoCAD arayüzünün birtakım özelleştirmelerle, katı nesnelerle çalışırken istenen sürükle ve bırak deneyimini sunma konusunda, daha elverişli olduğu tespit edilmiştir. Bu nedenle ikinci ve üçüncü uygulamalar için AutoCAD programı tercih edilmiştir. Ayrıca bu programla çalışmış olan öğrencilerin arayüz kullanım tercihleri dikkate alınmış; arayüzde yapmış oldukları iyileştirmelere dikkat edilmiştir. Bu kapsamda AutoCAD arayüzünde katılımcılara özgü yeni bir kullanıcı profili tanımlanmış; bu profilde nesne yakalama noktalarının açılması ve tanımlanması, modelleme araç çubuklarının içeriğinin özelleştirilmesi hususunda değişiklikler yapılmıştır.

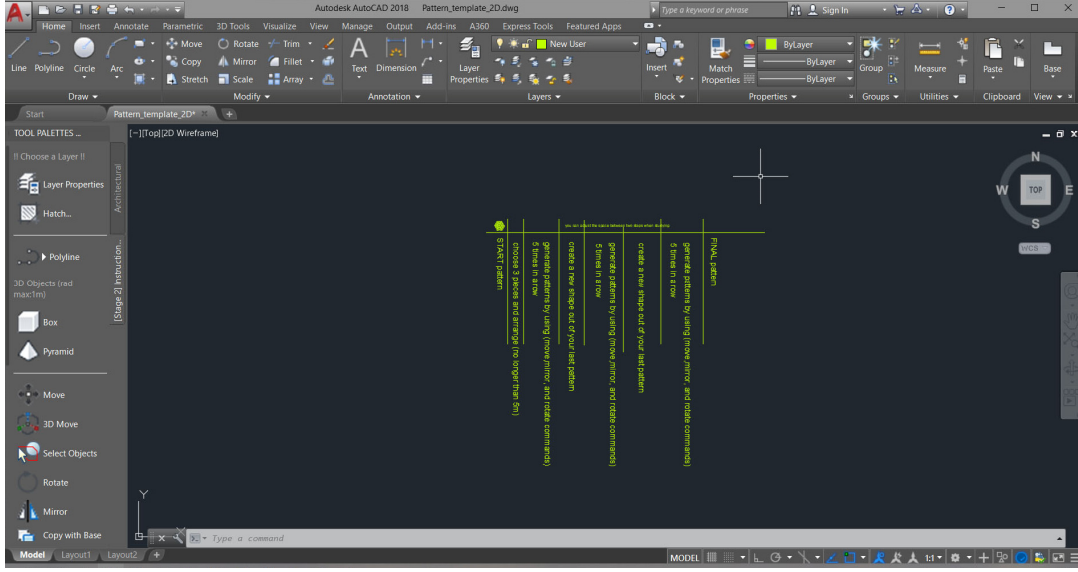
4.3.1.2 Uygulama 2: şekil yapma

İkinci uygulamada, öğrencilerin çizgi (wireframe) ve yüzey geometrilerle (surface) çalışırken değiştirdikleri algısal yaklaşımlarının sürece etkisini gözlemlemek amaçlanmıştır. Bu kapsamda, öğrencilere çizgi ve yüzey geometrilerle çalışmalarını gerektiren 2B Şekil Yapma görevi sunulmuştur.

Verilen görevde, öğrencilerden tanımlı kurallar çerçevesinde, bir örüntü tanımlamaları ve bu örüntüden seçtikleri çizgi ve yüzey geometrilerle yeni örüntüler oluşturmaları istenmiştir. Bunun için öğrencilere bir altlık tanımlanmış ve AutoCAD arayüzünde kullanmaları için bir belirli komutları içeren bir araç paleti (tool palette) oluşturulmuştur (Şekil 4.11).

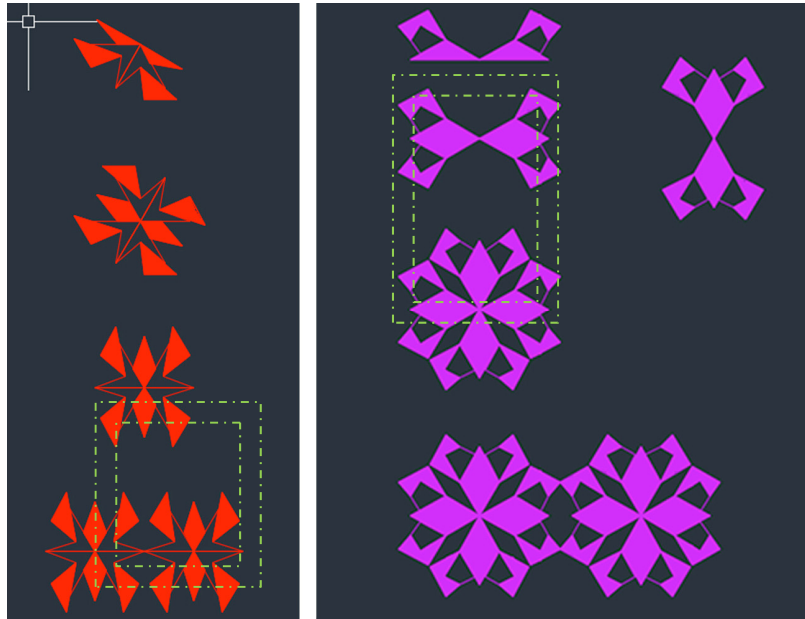
Hazırlanan altlıkla birlikte öğrencilere verilen talimatlar ve sınırlamalar, sırayla şu şekildedir:

- Altlıkta görmüş olduğunuz çizgi geometri üzerinde bir örüntü oluşturmak üzere <hatch> komutunu kullanın;
- Bir başlangıç modülü oluşturmak üzere oluşturduğunuz bu örüntüden iki dolu yüzey ve bir boş çizgi geometri seçin;
- Seçtiğiniz parçaları birbirleriyle çakışmayacak şekilde birleştirerek bir modül tanımlayın.
- Bu modülden, araç çubuğunda tanımlanmış kopyala/yapıştır <copy/paste>, aynala <mirror> ve döndür <rotate> komutlarını kullanmak koşuluyla toplamda beş hamle gerçekleştirerek bir örüntü türetin.
- Ortaya çıkardığınız bu örüntü içinden yeni bir dolu/boş geometri seçin ve mevcut örüntüye bu yeni modülü dahil etmek koşuluyla türetmeye devam edin.
- Bir önceki aşamada kullandığınız komutları, yine toplamda beş hamle gerçekleştirmek koşuluyla tekrarlayın.



Şekil 4.11 : Şekil Yapma görevi için oluşturulan araç paleti ve altlıktan bir ekran alıntısı.

Bu uygulama, öğrencilerin süreç içinde seçtikleri çizgi ve yüzey geometrileri, dolu ve boş olarak farklı algısal yaklaşımlarla nitelendirdiklerini ortaya çıkarmıştır. Öğrencilerin, örüntü oluşturma sırasında kullandıkları içi boş geometrileri, gerçekleştirdikleri birkaç hamleden sonra içi dolu geometriler olarak nitelendirdikleri gözlenmiştir (Şekil 4.12). Bu durumun ise verilen talimatları tekrarlayarak seri şekilde uygulamalarına engel teşkil ettiği saptanmıştır.



Şekil 4.12 : Öğrencilerin çizgi geometrileri farklılaşan algısal yaklaşımlarla ele alış biçimlerine dair iki farklı uygulama örneği.

Bu duruma binaen, yapılacak deney uygulamasında çizgi geometriler yerine yüzey geometrilerin kullanılması kararlaştırılmıştır.

Ayrıca, öğrencilerin karar alma ve uygulamada doğrusal olmayan iş akışları sergiledikleri saptanmıştır. Bu durum, uygulama sürecinde seçmiş oldukları komutları ve geometrilerle ilişkilendirmek üzere aldıkları kararlar arasında uzun süreli duraklamaları ortaya çıkarmıştır. Bu gözlemlerle, geleneksel CAD ile modelleme sürecinde gerçekleştirilen eylemlerin sistematik analizi için, tanımlanan görevlerin tasarım bağlamı dışında ve iyi-tanımlı problemler olarak nitelendirilmesinin uygun olduğu çıkarımı yapılmıştır.

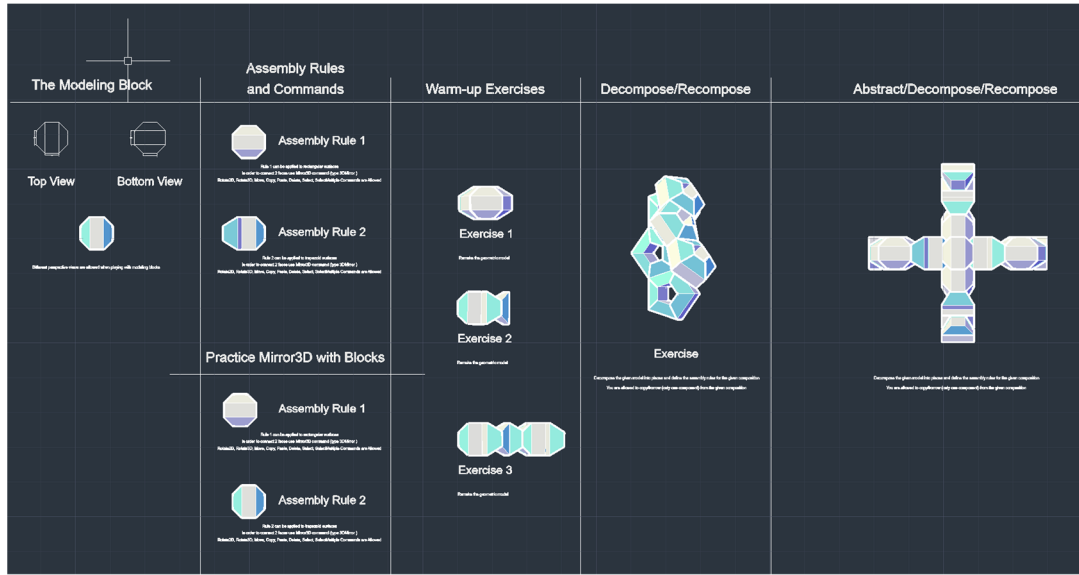
Dolayısıyla, son uygulamada, katılımcıların arayüz etkileşimlerinin sistematik analizi ve şekiller ve komutlar arasında kurdukları ilişkileri daha net okuyabilmek için, son uygulamanın iyi tanımlı bir problem niteliğinde kurgulanmış bir modelleme uygulamasına karar verilmiştir.

4.3.1.3 Uygulama 3: yüzeyler

Önceki uygulamada olduğu gibi, bu uygulama için de bir altlık oluşturulmuş ve öğrencilerin arayüz etkileşimlerine dair çıkarımlarda bulunmak amacıyla video-ekran kayıtları değerlendirmeye alınmıştır. Bu kapsamda, öğrencilerin modelleme sırasındaki komut kullanım stratejileri, seçtikleri komut ve şekillerdeki yaptıkları genellemeler dikkate alınmıştır. Uygulama, üç aşamadan oluşan iki farklı görevden oluşmaktadır.

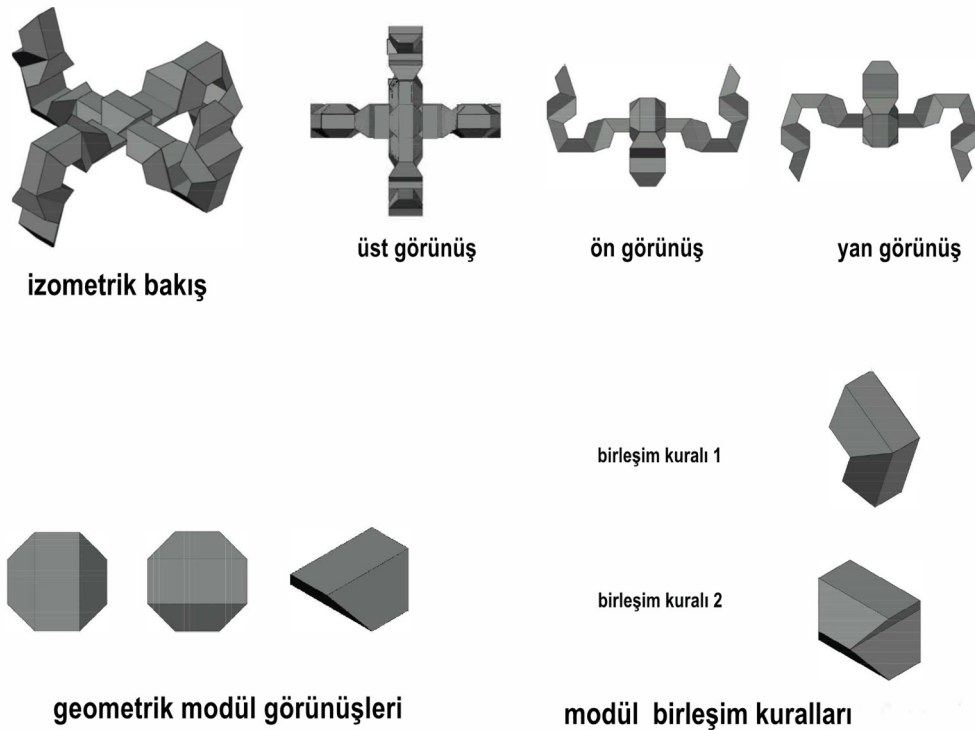
Verilen ilk görevin, içerik yönünden sonraki bölümde bahsedilecek olan deney uygulaması ile önemli benzerlikler taşıdığını söylemek mümkündür. Bu görevde öğrencilere iyi tanımlı bir problem olarak modelleme görevi verilmiş; verilen komutlar ve modülleri kullanmaları koşuluyla, öğrencilerden 3B bir geometrik şekli yeniden üretmeleri istenmiştir.

İlk aşamada, öğrencilerin verilen komut ve tanımlanan geometrik modüle aşinalık kazanmaları için çeşitli alıştırımlar tanımlanmıştır (Şekil 4.13). Bu alıştırımlarda, öğrencilere geometrik bir modülün <mirror3d> komutu ile türetilmesiyle elde edilen farklı şekiller verilmiştir. Ardından öğrencilerden, komut ve modüllerin tekrarlayan kullanımlarıyla oluşturulmuş bu şekilleri <mirror3d> komutunu kullanarak yeniden üreterek pratik yapmaları istenmiştir.



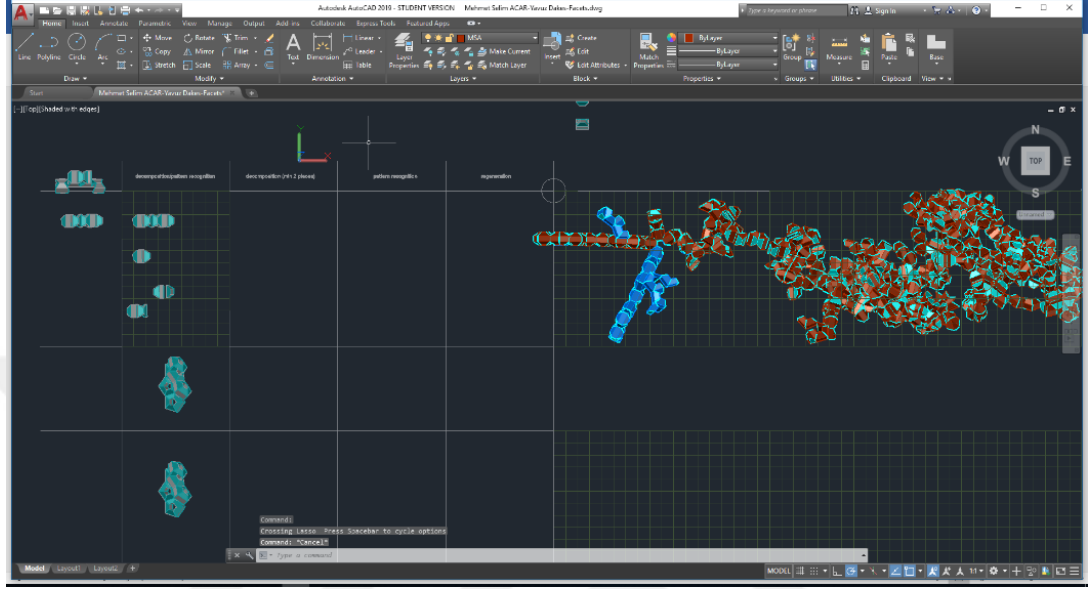
Şekil 4.13 : Yüzeyler görevi için tanımlanan uygulama altlığından bir ekran alıntısı.

Görevin ikinci aşamasında, uygulama olarak aynı modül ve iki farklı birleşim kurallarının tekrarlayan kullanımlarından türetilmiş bir şekil hazırlanmış ve öğrencilerden verilen bu şekli aynı komut ve birleşim kurallarıyla yeniden modellemeleri istenmiştir (Şekil 4.14).



Şekil 4.14 : Yüzeyler için tanımlanan ikinci görevde sunulan geometrik şekil.

İkinci görevde, öğrenciler ikili gruplara ayrılarak modül ve komutlarla modelleme sürecini bir oyun parkuruna taşımıştır. Bu oyunda, birbirlerine rakip oyuncular, dönüşümlü olarak kullanabildikleri <mirror3d> komutu ile verilen modülü türeterek parkuru tamamlamaya çalışmışlardır. (Şekil 4.15).



Şekil 4.15 : Yüzey Oyunu’nda oyuncuların hamlelerindeki şekil seçimlerine bağlı olarak karmaşıklıklaştan örüntüler oluşturmaya dönüşmesine bir örnek.

Yüzeylerin Modellenmesi ve Yüzey Oyunu için öğrencilere verilen talimatlar ve sınırlamalar sırayla şu şekildedir:

- Verilen geometrik modülü, tanımlanan birleşim kurallarını ve komutu dikkatlice inceleyin;
- (alıştırma 1 için) <mirror3d> komutunu kullanarak, altlıkta tanımlanan alanda örnekte sunulan şeklin aynısını üretin;
- (alıştırma 2 için) <mirror3d> komutunu kullanarak, altlıkta tanımlanan alanda örnekte sunulan şeklin aynısını üretin;
- (alıştırma 3 için) <mirror3d> komutunu kullanarak, altlıkta tanımlanan alanda örnekte sunulan şeklin aynısını üretin;
- (uygulama için) size tanımlanan şekli, en az iki modülden oluşacak biçimde parçalarına ayırın;

- (uygulama için) <mirror3d> komutunu kullanarak, altlıkta tanımlanan alanda parçaladığınız şeklin aynısını yeniden üretin.
- Oyunda size rakip olacak bir arkadaş seçin ve TeamViewer ile arkadaşınızın bilgisayarına bağlanın;
- Altlıkta sunulan oyun parkurunu dikkatlice inceleyin; parkurun sınır çizgilerine dikkat ederek verilen modülü <mirror3d> komutunu kullanarak türetin ve bu şekilde parkurun sonuna kadar ulaştırmaya çalışın.
- Gerçekleştirdiğiniz hamlelerle rakibinizin ilerlemesini engelleyebilirsiniz.
- Yaptığı son hamle ile parkur sonuna ilk ulaşan oyuncu oyunu kazanır.

Tanımlanan ilk görevde, tüm öğrencilerin aynı görevi ve uygulama araçlarını kullanmış olmaları, modelleme sürecinde komut-şekil seçimlerini karşılaştırılabilir düzeyde takip etmeye imkân vermiştir. Bu görev aynı zamanda, öğrencilerin modelleme stratejilerinde, gerçekleştirdikleri adımların sayı ve sıklıklarının yanında, şekil seçimi ve komut kullanımı yönünden yaptıkları ilişkilendirmeleri de karşılaştırmayı mümkün hale getirmiştir. Modelleme sürecinde, öğrencilerin verilen şekli ortaya çıkarmak için tanımladıkları modül birleşimleri ve komutlar arasında farklı genellemeler yaparak stratejiler geliştirdikleri gözlenmiş; bu genellemelerin şekil ve komut kullanımı yönünden, kavramsal ve prosedürel düzeyde soyutlamaları işaret ettikleri görülmüştür.

İkinci görevde ise, öğrencilerin hamle için seçtikleri modül veya şekiller üzerinde zaman zaman takılmalar (fixation) yaşadıkları görülmüştür. Bu kapsamda, ilk hamlelerinde aynı modüle takılan öğrenci gruplarının, ancak süreç sonuna doğru şekil seçimlerini zenginleştirdikleri ve daha karmaşık örüntüler elde ettiklerini belirtmek mümkündür.

Çalıştay kapsamında yer verilmiş tüm bu uygulamalar, deney içinde sunulacak görev içeriğini oluşturmaya ve değerlendirme yöntemini geliştirmeye çeşitli yönlerden yardımcı olmuştur. Bu kapsamda yapılan gözlemlerden elde edilen çıkarımları şu şekilde özetlemek mümkündür:

İlk çalışma sonucunda, AutoCAD ortamının arayüz kullanım kolaylığı ve analiz açısından elverişli olduğu saptanmış; ikinci çalışma sonucunda ise verilecek olan uygulamanın, tanımlanan komutlar ve geometrilerin tanımlı problemler çerçevesinde geliştirilmesi gerektiği görülmüştür. Son çalışmada ise, verilecek uygulamanın, içerik yönünden kavramsal ve prosedürel düzeyde soyutlamalar ve genellemeler yapmaya imkân verecek nitelikte olması gerektiğine dikkat edilmiştir. Bu kapsamda deneyde, geleneksel CAD programı kullanımı çerçevesinde, iyi-tanımlanmış problem niteliğinde hazırlanacak bir katı nesne modelleme uygulaması sunmanın elverişli olacağına karar verilmiştir.

4.4 Tasarımcı Bakış Açısıyla Hesaplamalı Düşünmede Soyutlama Becerisinin Değerlendirilmesi

Bireyin CAD ile modelleme sürecinde, komut kullanımı ve şekil seçimleri çerçevesinde yaptıkları soyutlamaları, kavramsal ve prosedürel yönden değerlendirmek üzere sistematik bir analize ihtiyaç duyulmuş; bu çerçevede farklı tasarım uzmanlığına sahip katılımcıların yer aldığı bir deney çalışması kurgulanmıştır. Pilot çalışmalarda olduğu gibi bu deneyde de katılımcılara CAD ortamında tanımlı komutlar ve modüllerin kullanılmasıyla elde edilen 2B bir geometriyi; yine aynı komut ve modülleri kullanarak modellemeleri için talimat verilmiş; bu kapsamda katılımcıların modelleme sürecinde gerçekleştirdikleri eylemler sistematik bir analiz çerçevesinde değerlendirilmeye alınmıştır.

Değerlendirme için katılımcıların seçtikleri şekiller ve uyguladıkları komutlar çerçevesinde yaptıkları genellemeler dikkate alınmıştır. Bu genellemelerin modelleme sürecine olan etkisini tespit etmek üzere, katılımcıların eylemleri, arayüz etkileşimleri, araç kullanımları ve görev tamamlama süreleriyle karşılaştırılmıştır. Bu doğrultuda, katılımcıların gereksiz tekrarlardan kaçınarak, modelleme sürecini mümkün olan en kısa sürede ve en az hamle ile gerçekleştirmelerine dikkat edilmiş; ancak, bu konuda katılımcılara herhangi bir açıklama yapılmamıştır.

Gerçekleştirilen bu deney için tanımlanan görev, değerlendirilme yöntemi ve süreç analizi aşağıda paylaşılmıştır.

4.4.1 Görev tanımı

Deneye katılan tüm katılımcıların modelleme sürecinde hem şekil hem de komut seçimlerini kolayca yapabilmeleri için, görevlerde tanımlanan modüller ve istenen şekillerin, tasarımcı olmayanlar tarafından da kolaylıkla uygulanabilir olmalarına özen gösterilmiştir. Bu nedenle, uygulama sürecinde, katılımcılara oyun blokları ile çalışmaya benzer, sürükle bırak deneyimi sunacak kolaylıkta bir ortam ve görev tanımlanmıştır. Bunun için, uygulamada verilen birim modüllerin hem uygulama kolaylığı açısından oldukça simetrik hem de uzamsal ilişkilerde çeşitliliğe izin verecek kadar açısız geometrilerden olmasına karar verilmiştir.

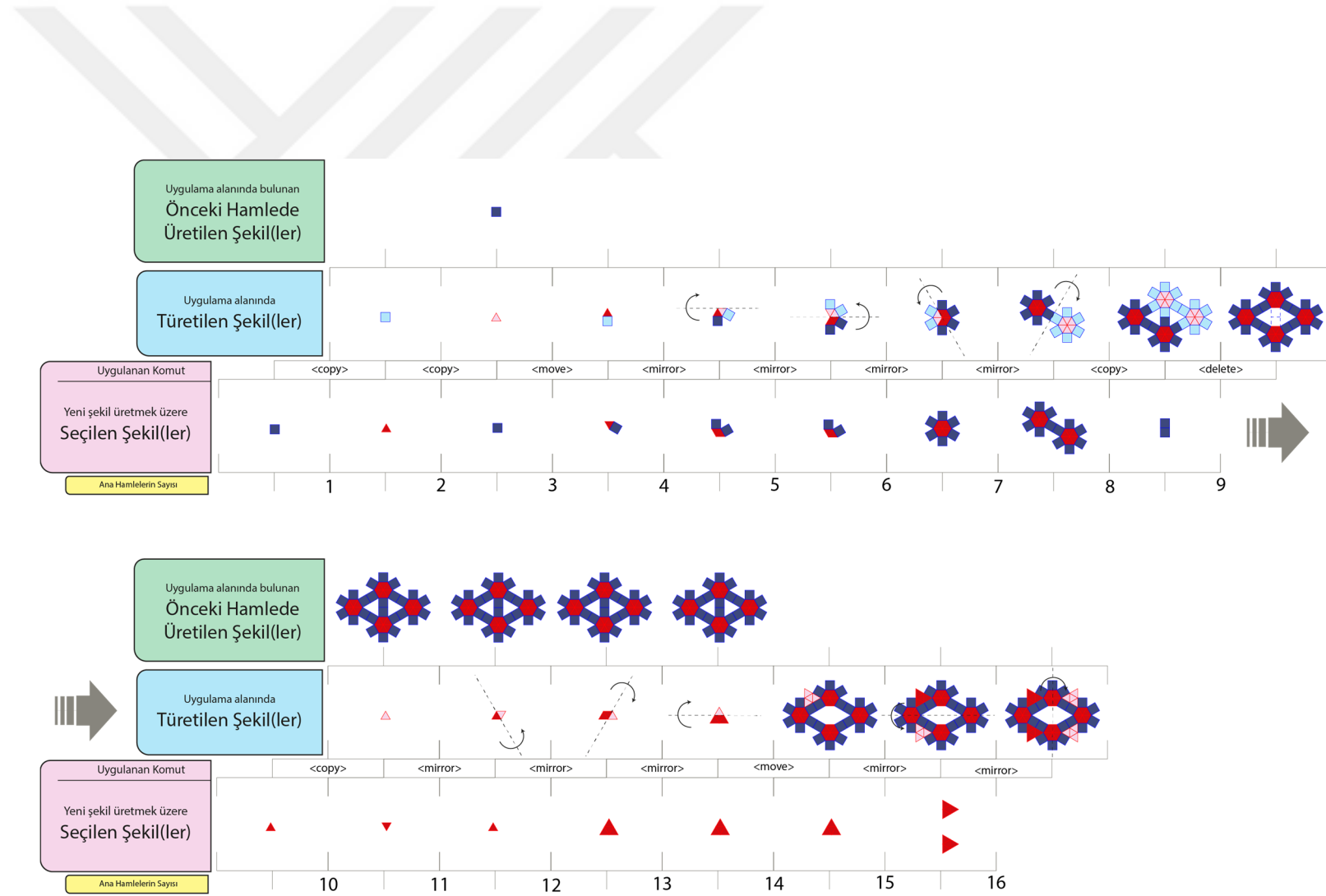
Ayrıca modüllerin, (X ve Y koordinatlarında) yatayda döndürüldüğünde veya görüş alanı (viewport) değiştirildiğinde, görünürlüklerinin aynı kalacak şekilde simetrik yapıda olmasına dikkat edilmiştir, uygulama 2B bir modelleme uygulaması olarak kurgulanmıştır.

Seçilen modüllerin, sadece yüzey kenarların çakışmalarına izin verecek şekilde birleştirilmelerine izin verilmiştir. Modüller arası bağlantı oluşturmak ve modülleri türetmek için kullanılabilecekler komutlar <copy/paste>, <move>, <delete> ve <mirror> olarak sınırlandırılmıştır. Bu komutları kullanan katılımcıların verilen komutları, birden fazla modüle uygulayabilmeleri için çoklu nesne seçimi (multiple object selection) yapmalarına izin verilmiştir. Bu şekilde katılımcıların modelleme sürecinde hem şekillerle yaptıkları gruplandırmalarını hem de prosedürel gruplandırmalarını görmek mümkün olmuştur. Bu tanımlama ve sınırlamalar, aynı zamanda deneyimli kullanıcıların izin verilen komutlar dışında herhangi bir komut veya araçla çalışmalarını da engellemeye yardımcı olmuştur.

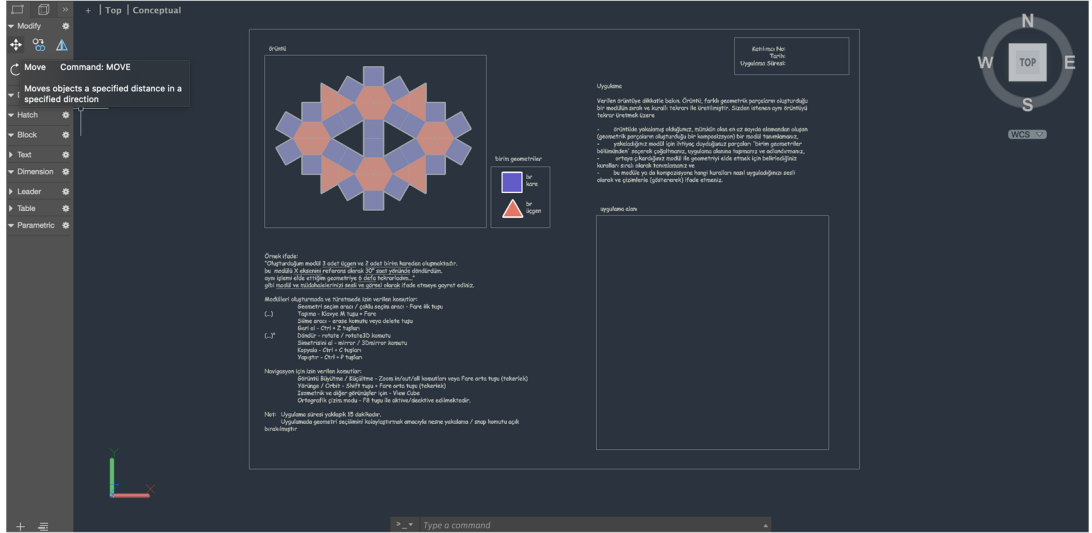
Katılımcılara modellemeleri için sunulan şekil, tanımlanan modüller ve bu modüllerin tekrarlayan komutlarla türetilmesinden oluşmaktadır. Sunulan şeklin modellenmesinde en az hamle ve komut kullanımı çerçevesinde oluşturulabilecek ideal çözüm önerilerinden biri Şekil 4.16'da gösterilmektedir. Bu şekil ile katılımcıların komut ve şekil seçimlerini, kavramsal ve prosedürel soyutlamalar ve genellemelere bağlı olarak gerçekleştirmeleri beklenmiştir.

Bu doğrultuda, pilot çalışmalardan elde edilen gözlemlere bağı olarak kullanıcı arayüzünde gerekli özelleştirmeler yapılmış; uygulama görevi için tanımlanan geometrik şekil, şekli türetmede kullanılan modüller ve izin verilen komutlar bir altlık ile AutoCAD’de sunulmuştur (Şekil 4.17). Ayrıca, verilen modüller ve komutların kullanımını kolaylaştırmak, bahsedilen sürükle ve bırak deneyimini sağlamak amacıyla, AutoCAD arayüzünde (ekran kısayollarının oluşturulması, geometrik modellemede köşe-kenar yakalama ayarlarının açılması gibi) çeşitli özelleştirmeler yapılmıştır.





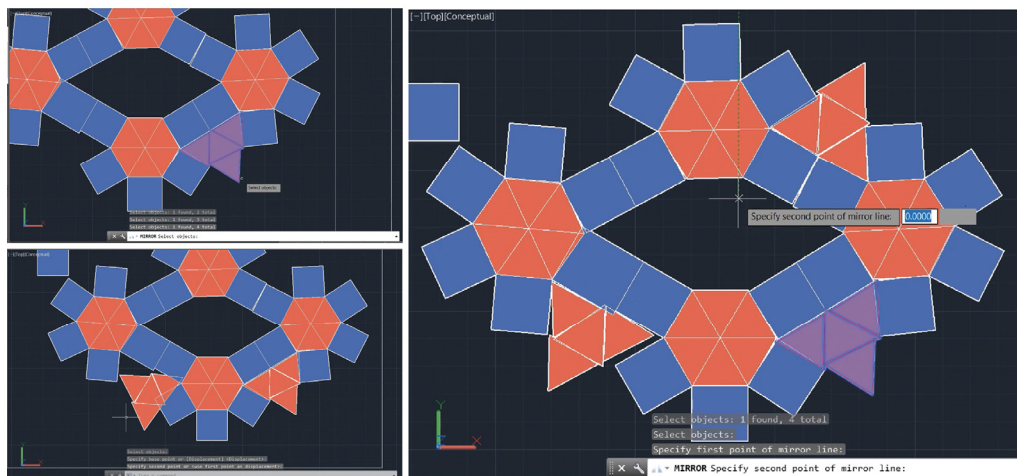
Şekil 4.16 : Tanımlanan geometrik şeklin minimum modellenmesi için kullanılabilecek hamle ve modül birleşimlerini gösteren ideal çözümlerlerden biri.



Şekil 4.17 : Uygulamada katılımcılara sunulan altlık ve uygulama arayüzünden ekran alıntısı.

4.4.2 Modül ve komutlarla alıştırma süreci

Deneylere başlamadan önce, AutoCAD arayüzü, geometrik modüllerin kullanımı, çerçevesinde, tüm katılımcılara tanımlanan uygulama araçları ve komutlarla pratik yapmaları için alıştırma süresi tanınmış; katılımcılara kendilerini hazır hissettikleri anda uygulamaya başlamaları için sözlü talimat verilmiştir. Tüm katılımcılar, verilen modüller ve komutları deneyimleyerek alıştırma sürecini başarıyla tamamlamıştır. Ancak katılımcıların performansları, arayüz kullanımı, araç hakimiyeti ve alıştırmaları tamamlama süresi yönünden kıyaslandığında, tasarım öğrencilerinin, tasarımcı olmayanlara göre belirgin biçimde daha başarılı olduklarını söylemek mümkündür. Nitekim, tasarımcı olmayan katılımcıların, arayüz araç kullanımındaki acemilikleri, uygulama sürecine farklı şekillerde yansımıştır (Şekil 4.18).



Şekil 4.18 : Tasarımcı olmayan katılımcılardan S6'nın uygulama sürecinden örnek.

4.4.3 Katılımcılar

Yapılan analizin içeriğinde, katılımcıların algısal yaklaşımlarının ve milisaniyelik aralıklarla gerçekleştirdikleri bilişsel eylemlerinin değerlendirmelerin yer alması nedeniyle, deneyde yer alan katılımcı sayısının toplamda 6 kişi ile sınırlandırılması uygun görülmüştür. Reisberg’e göre (2013) algısal yaklaşıma yer veren bu tür detaylı analizler için az sayıda katılımcı ile çalışmak mümkündür.

Bu kapsamda, yaş aralıkları 20-24 arasında değişen, 6. ve 8. yarıyıldan üç mimarlık öğrencisi ve bu öğrencilerle tasarım uzmanlığı yönünden farklılık göstermeleri için tasarım disiplinleri dışından seçilmiş (hukuk, tıp ve bilgisayar mühendisliği olmak üzere) üç lisans mezunu deneye katılmıştır. (Çizelge 4.2). Seçilen tüm katılımcılar, deneylerde gönüllülük esasına binaen yer almıştır. Öğrencilerin seçimi e-posta yoluyla yapılmıştır. Uygulama öncesi ve sonrasında, öğrencilerden gerçekleştirilen deney içeriği hakkında bilgilendirici olmamaları ve kimliklerini gizli tutmaları istenmiştir.

Çizelge 4.2 : Deneyde yer alan katılımcılarının demografik dağılımı.

Tasarım Uzmanlığı	Katılımcı Adı	Cinsiyet	Yaş Aralığı	Dijital Araç Kullanımı	Eğitim Alanı	Eğitim Durumu
				Programlama veya CAD/CAM/CAE Eğitimi		
Yetkin	S1	Kadın	20-24	Var – Tasarım	Mimarlık	Öğrenci (4.sınıf)
	S2	Kadın	20-24	Var – Tasarım	Mimarlık	Öğrenci
	S3	Kadın	20-24	Var – Tasarım	İç Mimarlık	Öğrenci (3.sınıf)
Deneyimsiz	S4	Erkek	20-24	Var - Programlama	Mühendislik	Mezun
	S5	Erkek	20-24	Yok	Hukuk	Yeni Mezun
	S6	Kadın	20-24	Yok	Tıp	Mezun

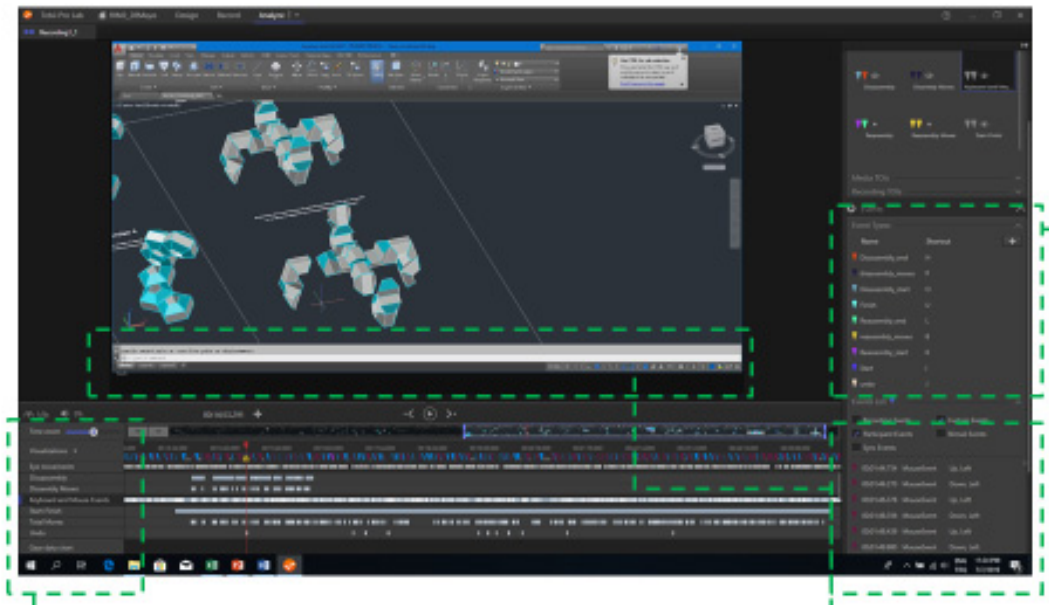
Pilot çalışmalarda olduğu gibi, deney için de seçilen mimarlık öğrencilerinin geçmişte profesyonel düzeyde (ders, çevrimiçi dersler veya özel kurs aracılığıyla) modelleme veya programlama eğitimi almamış olmasına dikkat edilmiştir.

Ayrıca, uygulama içeriği, deney ortam ve araçlarının kullanımı gereği tüm katılımcılar, uzamsal ve motor becerilerin yetkinlikleri çerçevesinde 6 sorudan oluşan mini bir teste tabi tutulmuştur (EK B).

Uygulanan bu testte, katılımcılara ekran üzerinde görsel algı, zihinsel döndürme, yön belirleme, arayüz araçlarını kullanım becerilerini hedef alan sorular yöneltilmiş; arayüz komutlarının kullanımı için İngilizce bilgileri dikkate alınmıştır. Tüm katılımcılar sunulan soruları zorluk çekmeden yanıtlamış; katılımcıların iki boyutlu ortamda yön kullanımlarında bir soruna rastlanmamıştır. Ayrıca, görsel algı sorularına vermiş oldukları yanıtlarda da belirgin farklılıklara rastlanmamıştır.

4.4.4 Veri toplama yöntemi ve kullanılan araçlar

Bu sistematik analiz çerçevesinde veri olarak, tüm katılımcıların ekran üzerindeki eylemleri (video-ekran görüntüleri, göz hareketleri, program komut geçmişi betiği ve arayüz etkileşimleri) ve araç kullanımları kayıt altına alınmıştır. Katılımcıların göz hareketleri ve arayüz etkileşim kaydı için Tobii Pro X2-30 ekrana bağlı göz takip cihazı, analizi için ise Tobii ProLab deneme sürümü kullanılmıştır (Şekil 4.19). Bu kayıtlar, katılımcıların gerçekleştirdikleri eylemlerin ortaya çıkardığı kavramsal ve prosedürel soyutlama örüntülerini saptamaya yardımcı olmuştur.



Şekil 4.19 : Tobii ProLab Kullanıcı arayüzünden bir ekran alıntısı.

4.4.5 Değerlendirme yöntemi

Her bir katılımcının modelleme süresini tanımlamak için, Tobii ProLab’da kayıt altına alınan ekran video kayıtları üzerinde işlenmek üzere, görevine başlama ve görevi tamamlama sürelerinin işaretlendiği bir zaman çizelgesi (timeline) oluşturulmuştur.

Modelleme süresi, katılımcının uygulama alanı üzerindeki ilk göz hareketinden, görevi tamamlamak için gerçekleştirdiği son eyleme kadar geçen süreyi tanımlanmıştır.

Ardından bu çizelge üzerinde, katılımcıların türetmek için kullandıkları şekilleri seçimleri, ve bu şekli türetme anları, ekran üzerindeki göz hareketleri ve fare kullanımları ile eşleştirilerek işaretlenmiştir. Bu doğrultuda, katılımcıların verilen geometrik modeli elde etmek üzere kullandıkları modüller, modüllerle türettikleri şekiller ve bu şekilleri türetmek için kullandıkları komutlar tespit edilmiş; zaman çizelgesi üzerinde işaretlenmiştir, Daha sonra, bu sistematik analize çerçevesinde elde edilen bulgular. Tüm katılımcıların modelleme süresince kullandıkları modül/şekiller, modül/şekil türetme yöntemleri ve türettikleri şekiller grafiklerle EK C’de sunulmuştur. Deney kapsamında oluşturulan kodlama şeması ve tanımlanan segment değerleri çerçevesinde değerlendirmeye alınmıştır.

4.4.5.1 Kodlama şemasının oluşturulması

Kodlama şeması, katılımcıların modelleme süresince gerçekleştirdikleri eylemlerin, türü ve içeriğine bağlı olarak oluşturulmuştur. Kodlamada dikkate alınan bu eylemlerin kategorizasyonunda dikkate alınanları Şekil 4.20’deki gibi özetlemek mümkündür.

UYGULANAN EYLEM TÜRÜ	EYLEMİN İÇERİĞİ	EYLEMİN DURUMU
ANA HAMLE	<move>	KABUL EDİLEN HAMLE
ARA HAMLE	<copy>	REDDİLEN HAMLE
	<mirror>	
	<delete>	

Şekil 4.20 : Kodlama şeması için tanımlanan değerler.

Adım-1: uygulanan eylemin türünün tanımlanması

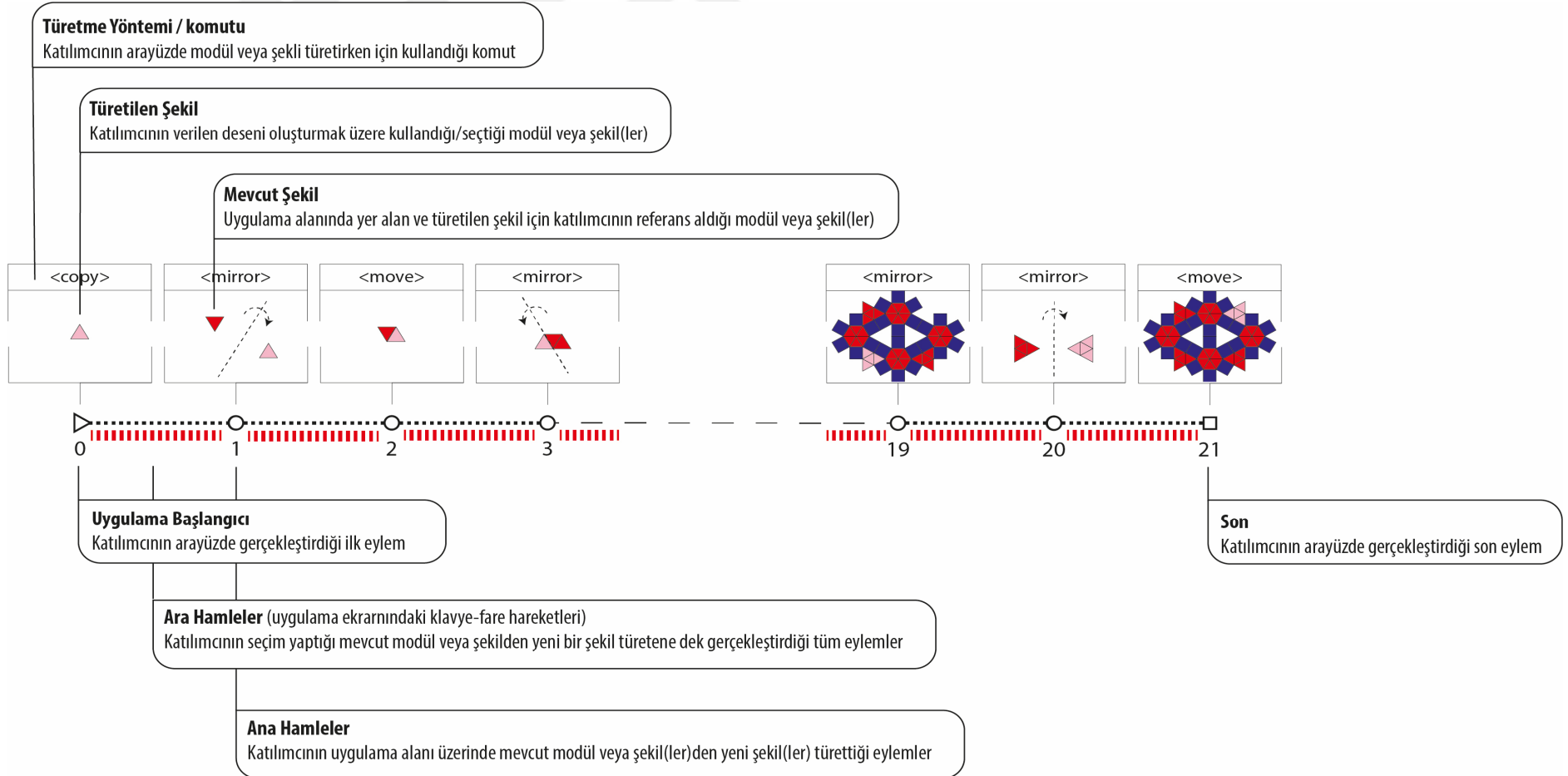
İlk olarak, zaman çizelgesine sadık kalınarak, katılımcıların mevcut şekilden yeni bir şekil türetmek üzere gerçekleştirdikleri her türlü eylem, içeriğine bakılmaksızın ‘ana hamle’ olarak nitelendirilmiştir (Şekil 4.21). Ardından, Tobii ProLab arayüzünde kayıt altına alınmış olan, klavye ve fare kullanımıyla gerçekleştirilmiş her türlü eylem, ‘ara hamle’ olarak nitelendirilmiştir. Bu bağlamda, geometrik modelin ortaya çıkarılması için için kritik olan ana hamleler, ara hamlelerden ayrıştırılmıştır.

Adım-2: uygulanan eylemin içeriğinin tanımlanması

Bu aşamada, tespit edilen ana hamleler, katılımcının seçtiği modül/şekil ve türetmek için seçtiği komut (move, delete, copy, mirror) çerçevesinde kategorize edilmiş; ancak segment değerlerinin tanımlanması sırasında renkli olarak kodlanmıştır.

Adım-3: uygulanan eylemin durumunun tanımlanması

Son olarak, kategorize edilen tüm ana hamleler, katılımcıların modelleme sürecindeki uygulamalarına bağlı olarak olarak ‘kabul edilen’ veya ‘reddedilen’ olmak üzere ayrıştırılmıştır. Eğer katılımcı, gerçekleştirdiği ana hamle için herhangi bir geri alma veya silme işlemi uygulamışsa, bu hamle ‘reddedilen’ bir eylem olarak nitelendirilmiş; aksi takdirde ‘kabul edilen’ bir eylem olarak nitelendirilmiştir.



Şekil 4.21 : Modelleme sürecinde katılımcıların eylemlerinin analizinde zaman çizelgesi üzerinde tanımlanan değerlerin grafik açılımı.

4.4.5.2 Segment değerlerinin tanımlanması

Oluşturulan kodlama şemasıyla, katılımcıların modelleme sürecinde hangi komutları ve şekilleri hangi sırayla kullandıkları ve hangi ara şekilleri türettikleri tespit edilmiştir. Bu tespit aynı zamanda, gerçekleştirilen ana hamlelerin içerik yönünden ne derece benzerlik taşıdıklarını görmeye ve katılımcıların modelleme süreçlerinde gereksiz tekrarlara ne kadar yer verdiklerini saptamaya yardımcı olmuştur. Dolayısıyla, değerlendirme için segment değerleri de bu aşamada tanımlanmıştır.

Katılımcıların performanslarını karşılaştırmalı olarak değerlendirmek için, gerçekleştirdikleri ana hamleler, kodlama şemasına bağlı olarak nitelendirilmiştir. Bu çerçevede, her bir hamle içeriğine göre bir ‘segment’e dönüştürülmüştür. Aynı kodlama içeriklerine sahip ve arka arkaya gereksizce tekrar eden segmentler ise gruplandırılarak ‘çoklu segmentler’ olarak adlandırılmıştır. Her bir katılımcının modelleme sürecine ait segmentler, Şekil 4.22’de görüldüğü üzere numaralandırılmıştır.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
<copy>	<mirror>	<copy>	<move>	<mirror>	<copy>	<mirror>	<mirror>	<move>	<mirror>	<mirror>	<mirror>	<mirror>	<delete>
Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted	Accepted
1	2	3	4	5	6	7	8	9	10	11			12

segment
örneği

çoklu segment
örneği

Şekil 4.22 : S1’in ana hamlelerindeki kodlama içeriğine bağlı olarak tanımlanan ‘segment’ ve ‘çoklu segmentler’den bir örnek.

Segmentlerin gruplandırılması, katılımcıların modelleme süreçlerinde gerçekleştirdikleri kavramsal ve prosedürel soyutlama örüntülerini ortaya çıkarmıştır. Katılımcıların soyutlama becerilerinin değerlendirilmesi ise, modelleme süreçleri için tanımlanmış olan çoklu segment sayısının, toplam segment sayısının ve sıklığının, toplam ana hamle sayılarıyla karşılaştırılması çerçevesinde gerçekleştirilmiştir. Bu karşılaştırma, katılımcıların prosedürel soyutlama becerilerini nicel yönden kıyaslamaya yardımcı olmuştur. Ancak hesaplamada, reddedilen ana hamleler dikkate alınmamıştır. Tüm katılımcıların modelleme süreçlerinde gerçekleştirdikleri hamleler, kodlama şemaları ve tanımlanan segment değerleri Şekil 4.23’te grafik olarak verilmiştir.





4.5 Değerlendirme

Oluşturulan kodlama şeması ve şemaya bağlı olarak tanımlanan segment değerleri çerçevesinde ortaya çıkarılmış olan Şekil 4.23'te yer alan bilgilerin, katılımcıların ara hamle değerleri ve görev tamamlama süreleriyle birlikte karşılaştırıldığı bu değerlendirmede, katılımcıların performanslarını

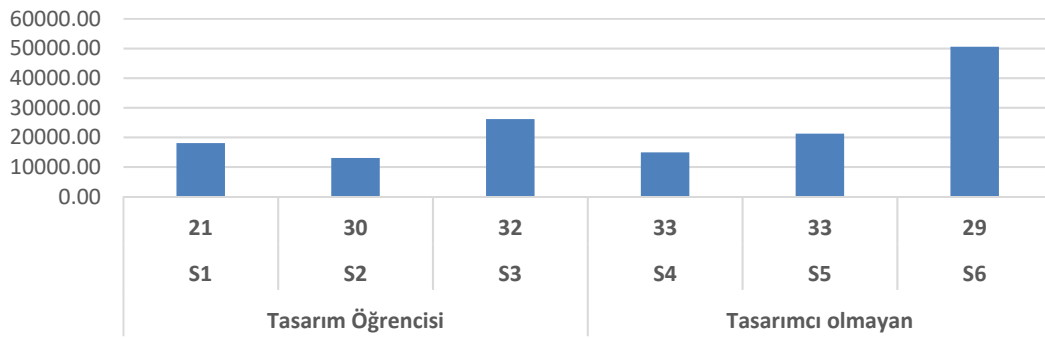
- Görsel hesaplamada kavramsal ve prosedürel soyutlama ilişkisi,
- Tasarım uzmanlığının kavramsal ve prosedürel soyutlamaya etkisi,
- Arayüz etkileşimi ve araç kullanımında tasarım uzmanlığının rolü,

olmak üzere üç başlık altında yorumlamak mümkündür.

4.5.1 Görsel hesaplamada kavramsal ve prosedürel soyutlama ilişkisi

Bu kapsamda, öncelikle katılımcıların gerçekleştirdikleri ana hamle sayıları karşılaştırdığımızda, en başarılı katılımcının S1, en zayıfın ise S6 olduğu görülmektedir. Ancak gerçekleştirilen ara hamle sayılarını katılımcıların görev tamamlama süreleri çerçevesinde karşılaştırdığımızda, milisaniye başına gerçekleştirilen hamle sayısı yönünden, en hızlı işlem yapan katılımcının S2, en yavaşın S3 olduğunu söylemek mümkündür (Çizelge 4.3).

Çizelge 4.3 : Katılımcıların her bir ana hamle için harcamış olduğu ortalama süre (ms)



Tasarım öğrencisi (S1, S2 ve S3) ve tasarımcı olmayan (S4, S5 ve S6) katılımcıların uygulama sürecinde gerçekleştirdikleri ara ve ana hamle sayıları, görev tamamlama süreleri ve ortalama değerleri ise Çizelge 4.4'te yer almaktadır.

Çizelge 4.4'te, genel ortalamaya baktığımızda, tasarım öğrencilerinin (S1, S2 ve S3), tasarımcı olmayanlara kıyasla daha hızlı çalıştıkları ve daha az ana hamle ile görevi tamamladıkları görülmektedir. Öte yandan, her iki katılımcı grup içinde ortalama dan uzakta performans gösteren katılımcılar mevcuttur. Bu çizelgeye göre tasarım öğrencilerinden S3, kendi grup ortalaması altında kalırken; tasarımcı olmayanlardan S4, kendi grup ortalamasının üstünde performans göstermiştir. Bu durum, araç kullanım becerisinin araçla işlem yapma becerisinden farklı olarak değerlendirilmesi gerektiğini ispatlar niteliktedir.

Çizelge 4.4 : Katılımcıların görev tamamlama süreleri ve gerçekleştirdikleri hamle değerleri

Tasarım Uzmanlığı	Katılımcı Adı	Toplam Ana Hamle Sayısı	Kabul Edilen Ana Hamle Sayısı	Toplam Ara Hamle Sayısı	Görev Tamamlama Süresi (ms)	Ana Hamle İçin Harcanan Ortalama Süre (ms)	Ara Hamle İçin Harcanan Ortalama Süre (ms)
Tasarım Öğrencisi	S1	21	21	442	379540	18073.33	858.69
	S2	30	23	722	390722	13024.07	541.17
	S3	32	31	691	837312	26166.00	1211.74
Tasarım Öğrencileri	Ortalama Değer	27.67	25.00	618.33	535858.00	19087.80	870.53
Tasarımcı olmayan	S4	33	33	511	492863	14935.24	964.51
	S5	33	33	633	701779	21266.03	1108.66
	S6	29	29	968	1467107	50589.90	1515.61
Tasarımcı olmayanlar	Ortalama Değer	31.67	31.67	704.00	887249.67	28930.39	1196.26
Tüm Katılımcılar	Ortalama Değer	29.67	28.33	661.17	711553.83	24009.09	1033.39

Çizelge 4.5'te yer alan katılımcıların gerçekleştirmiş oldukları ana hamlelerindeki benzerlik ve farklılıklara bağlı olarak tanımlanmış olan segment değerlerine bakıldığında, toplam segment sayısının kabul edilen ana hamle sayısına oranıyla en başarılı katılımcının (programlama deneyimi olan) S4 olduğu görülmektedir.

Çizelge 4.5 : Hamlelerin içeriğindeki tekrarlara göre tanımlanan segment değerleri ve yüzdelik dağılımı.

Tasarım uzmanlığı	Katılımcı adı	Kabul Edilen Ana Hamle Sayısı	Tanımlanan Toplam Segment Sayısı	Çoklu Segment Sayısı	Çoklu Segmentlerde Yer Alan Ana Hamle Sayısı Ortalama Değeri	Çoklu Segmentlerde Yer Alan Toplam Ana Hamle Sayısı	Tanımlanan Segment Sayısı Kabul Edilen Ana Hamle Sayısına Yüzdelik Oranı
Tasarım Öğrencisi	S1	21	17	2	3	6	81%
	S2	23	18	4	2.25	9	78%
	S3	31	20	4	3.75	15	65%
Tasarım Öğrencileri Genel Ortalama		25	18.33	3.33	3	10	75%
Tasarımcı olmayan	S4	33	28	5	2	10	85%
	S5	33	24	7	2.29	16	73%
	S6	29	22	6	2.17	13	76%
Tasarımcı olmayanlar Genel Ortalama		31.67	24.67	6	2.15	13	78%
İdeal Çözüm Önerisi		16	14	4	2	4	88%

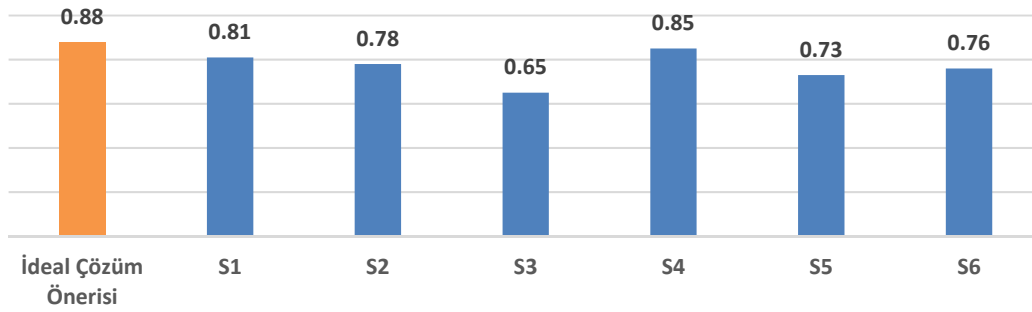
Şekil 4.23'te sunulmuş olan segment değerlerine bakıldığında, bu katılımcının, çoklu segmentleri içinde en fazla iki defa tekrara yer verdiği ve bu çerçevede ideal çözüm önerisine oldukça yakın bir değer elde ettiği görülmektedir. Bu bağlamda S4'ün çoklu segmentlerde gereksiz tekrarlardan kaçınmış olması, katılımcının prosedürel soyutlama çerçevesinde daha başarılı olduğunu göstermiştir. Yine Çizelge 4.5'te, çoklu segmentlerin içerdiği tekrarlayan hamle sayılarına bakıldığında, en az sayıda tekrar yapanın %85 ile S4 olduğu görülmektedir.

Bu kapsamda S4'ten sonra, tasarım öğrencileri S1'in %81 ve S2'nin %78 ile ideal çözüm önerisi için tanımlanmış %85'e oldukça yakın olduklarını söylemek mümkündür. Ancak S3 ise tüm katılımcılar arasında en düşük yüzdeyle görevi tamamlamıştır. Bu nedenle, tasarım uzmanlığının prosedürel soyutlamaya etkisinden söz etmek mümkün olmasa da prosedürel soyutlamanın görsel hesaplamada belirleyici rol oynadığını belirtmek mümkündür.

4.5.2 Tasarım uzmanlığının kavramsal ve prosedürel soyutlamaya etkisi

Çizelge 4.6’da ideal çözüm önerisi ve katılımcıların segment değerlerinin kabul edilen hamle sayısına yüzdelik oranı dikkate alındığında, en başarılı katılımcının S4 olduğu görülmektedir. Çizelgedeki ideal çözüm önerisi ve katılımcıların segment değerlerinin kabul edilen hamle sayısına yüzdelik oranı dikkate alındığında, en başarılı katılımcının S4 olduğu görülmektedir.

Çizelge 4.6 : Katılımcıların kavramsal ve prosedürel soyutlama becerilerinin kıyaslanması.



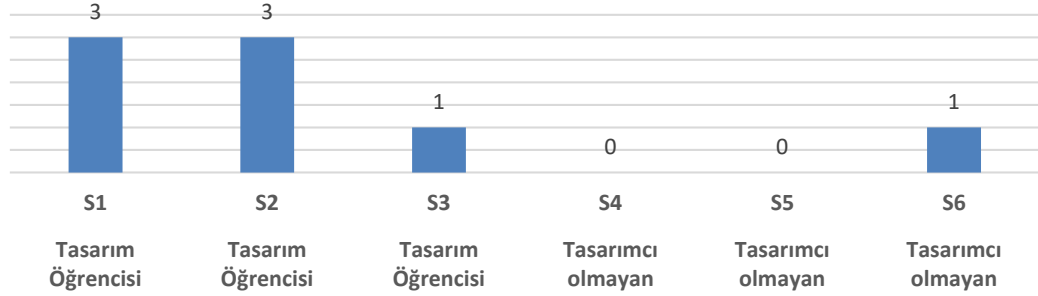
S3’ün kendi seviyesindeki tasarımcılara göre daha düşük performans göstermesinin bir nedenini, uygulama sürecinde izlediği stratejiye bağlamak, seçtiği komut ve şekilleri ele alış biçimi ile ilişkilendirmek mümkündür. S3’ün performansındaki bu farklılığa, Pea ve Kurland’in (1984) acemi programcıları gözlemiş oldukları çalışmalarında rastladıkları “bilgisayarda anlayış geliştirmeden üretmek” davranışının neden olduğu söylenebilir (s:151). Bu katılımcının özellikle modülleri çoğaltırken, komut ve şekil seçimi yönünden takılmalar göstermesi (Çizelge 4.7), süreç içinde de diğer tasarım öğrencilerine kıyasla daha zayıf bir stratejiyle çalışmasına neden olmuştur.

Çizelge 4.7 : Modül veya şekil türetmek için katılımcıların kullanmış oldukları komutlar ve kullanım sıklıkları.

		<copy>	<move>	<mirror>	<delete>
Tasarım Öğrencisi	S1	4	3	13	1
	S2	8	4	13	6
	S3	10	6	14	2
Tasarımcı olmayan	S4	4	12	16	2
	S5	12	11	8	2
	S6	5	7	17	0

Çizelge 4.8’de katılımcıların ana hamlelerinde kullandıkları modüller ve türetmiş oldukları şekillerin çeşitliliğine ve kullanım sıklığına bakıldığında ise, modül türetme konusunda tasarım öğrencilerinin daha başarılı olduğunu belirtmek mümkündür.

Çizelge 4.8 : Katılımcıların ana hamleler içinde türettikleri ve kullandıkları orijinal şekil sayısı.



Özellikle S1 ve S2’nin, ana hamlelerinde kullandığı tekli modül sayısının diğer katılımcılarınkinden daha az olduğu görülmektedir. S1 ve S2 modüllerle yeni şekil türetme konusunda diğer tüm katılımcılardan daha başarılı olmuştur. Bu çerçevede, tasarım öğrencilerinin türettikleri şekillerin çeşitliliği, tasarımcı olmayanların türettiklerinden daha fazla olduğu görülmektedir. Diğer yandan, S3’ün tasarımcı olmayanlara (S4, S5 ve S6) benzer biçimde ağırlıklı olarak tekil modüllerle işlem yaptığı; bu nedenle, diğer tasarım öğrencileri kadar ara şekil yönünden çeşitlilik sunmadığı görülmektedir.

Nitekim, Çizelge 4.8’de görüldüğü üzere, S6 modüllerle özgün bir şekil türetmiş ve uygulama sürecinde kullanabilmiştir. Dolayısıyla, türetilen şekillerin özgünlüğüne bakarak, katılımcıların kavramsal düzeyde soyutlama becerisini tasarım uzmanlıkları ile ilişkilendirmek oldukça zordur. Katılımcıların görev süresince komutlarla ilişkilendirerek türettikleri ve kullandıkları ara şekiller ve kullanım sıklıklarına ait ortalama değerler Çizelge 4.9’da gösterilmiştir. Çizelgede katılımcıların ana hamlelerinde kullandıkları modüller ve türetmiş oldukları şekillerin çeşitliliğine ve kullanım sıklığına bakıldığında, modül türetme konusunda tasarım öğrencilerinin daha başarılı olduğunu belirtmek mümkündür. Ancak, S3’ün tasarımcı olmayanlara (S4, S5 ve S6) benzer biçimde ağırlıklı olarak tekil modüllerle işlem yaptığı; bu nedenle, diğer tasarım öğrencileri kadar çeşitli ara şekil türetmemiş olduğu görülmektedir. Diğer yandan, S6’nın özgün şekil üretmiş ve uygulama sürecinde kullanmış olması, tasarımcıların kavramsal soyutlama becerisini tasarım uzmanlıkları ile ilişkilendirmeyi zorlaştırmaktadır.

Çizelge 4.9 : Katılımcıların görev süresince komutlarla ilişkilendirerek türettikleri ve kullandıkları ara şekiller ve kullanım sıklıkları.

														
Tasarım Öğrencisi	S1	6	0	3	0	2	1	2	1	3	0	0	0	3
	S2	8	5	0	0	6	0	3	0	0	2	2	0	4
	S3	12	0	0	0	5	0	10	1	0	0	0	0	3
Tasarımcı olmayan	S4	10	0	0	1	7	0	11	0	0	0	0	0	4
	S5	8	0	0	1	5	0	13	0	0	0	0	0	6
	S6	8	0	0	1	5	0	10	0	0	0	0	2	3
Ortalama Değer		8.67	0.83	0.50	0.50	5.00	0.17	8.17	0.33	0.50	0.33	0.33	0.33	3.83

4.5.3 Arayüz etkileşimi ve araç kullanımında tasarım uzmanlığının rolü

Katılımcıların araç kullanım becerileri karşılaştırıldığında, sonuçların oldukça tahmin edilebilir olduğunu söylemek mümkündür. Çizelge 4.10'da katılımcıların görev tamamlama süresince gerçekleştirdikleri eylemler, hamle sayıları ve araç kullanımlarına ait nicel değerlere bakıldığında, tasarım öğrencilerinin daha başarılı oldukları görülmektedir.

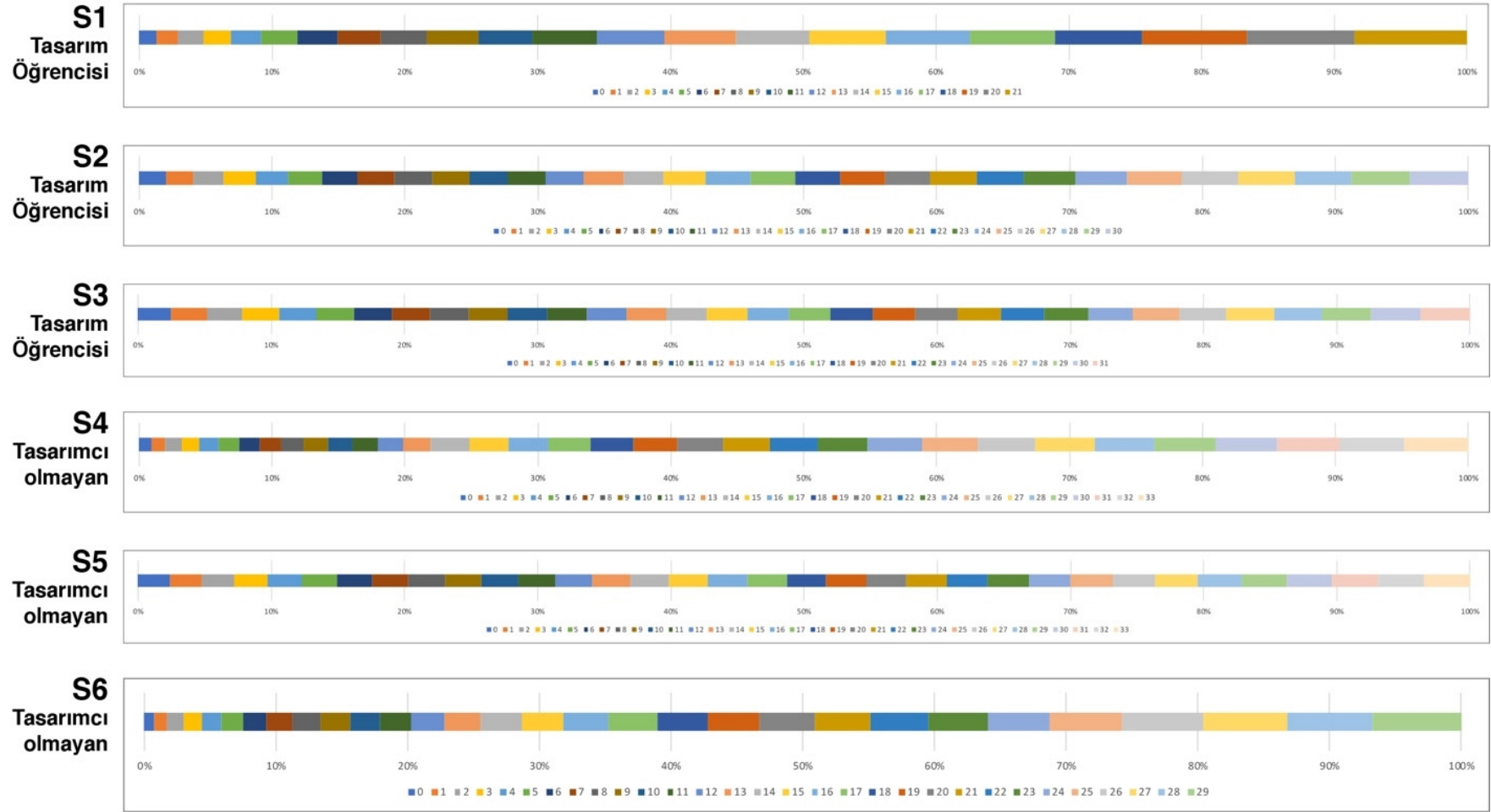
Yine aynı çizelgede, katılımcıların araç kullanımları karşılaştırıldığında, tasarımcı olmayanların arayüzde yabancılik çekmeleri nedeniyle, modelleme sürecinde daha fazla ara hamle gerçekleştirmiş oldukları görülmektedir. Bu durumu etkileyen faktörlerden biri, tasarımcı olmayan katılımcıların, nesneleri döndürürken veya taşıırken sayısal değerler kullanmamasından kaynaklanmaktadır. Yine, Çizelge 4.10'da katılımcıların klavye ve fare kullanımlarına bakıldığında, özellikle tasarımcı olmayan katılımcıların, eylemlerini ağırlıklı olarak ekran üzerindeki araç çubukları aracılığıyla gerçekleştirdikleri görülmüştür. Nitekim, S6'nın modelleme sırasında komut seçimlerini, ekran yer alan araç çubuğu ile gerçekleştirmiş ve klavyeyi neredeyse hiç kullanmamış zorlanmış olması bu durumun önemli bir göstergesidir. Diğer yandan tasarımcı olmayan S4 ve S5'in, klavye kullanmada, S6 kadar zorlanmadığı görülmüştür.

Çizelge 4.10 : Katılımcıların arayüz ile etkileşimi süresince gerçekleştirdikleri eylemler ve görev tamamlama sürelerinin karşılaştırılması.

Tasarım Uzmanlığı	Katılımcı Adı	Gerçekleştirilen Ana Hamle Sayısı	Ara Hamlelerde Klavye Kullanımı	Ara Hamlelerde Fare Kullanımı	Arayüz Etkileşimindeki Klavye-Fare Kullanımı	Görev Tamamlama Süresi (ms)
Tasarım Öğrencisi	S1	21	144	298	442	379540
	S2	30	148	574	722	390722
	S3	32	139	552	691	837312
Tasarım Öğrencileri	Genel Ortalama	27.67	143.67	474.67	618.33	535858.00
Tasarımcı olmayan	S4	33	95	416	511	492863
	S5	33	59	574	633	701779
	S6	29	4	964	968	1467107
Tasarımcı Olmayan	Genel Ortalama	31.67	52.67	651.33	704.00	887249.67
Tüm Katılımcılar	Genel Ortalama	29.67	98.17	563.00	661.17	711553.83

Çizelge 4.10'da, tasarımcı olmayanların (S4, S5 ve S6) performanslarının ortalama değerlerinin, tasarım öğrencilerine (S1, S2 ve S3) kıyasla belirgin farklılık gösterdiği görülmektedir. Bu durumun, literatürde Rosson (1983), Nilsen ve diğ.'nin (1993) araç etkileşiminde uzmanlığın rolü üzerine gerçekleştirdikleri araştırma sonuçları ile benzerlik taşıdığı söylenebilir. Rosson (1983), Nilsen ve diğ. (1993) çalışmalarında, acemi bireylerin ofis yazılımı kullanmada pratik yaparak becerilerini geliştirdiklerini gözlemlemiş; ancak uzman kullanıcılarının komut seçme ve klavye kullanma konusunda, deneyim kazanan acemi bireylere göre daha bilinçli olduklarını saptamışlardır. Bu uygulamada da tasarım öğrencileri AutoCAD arayüzünde önceden deneyim sahibi olmaları sebebiyle, tasarımcı olmayanlara göre eylemlerini daha hızlı gerçekleştirmiş; özellikle S1 ve S2 görevi daha kısa sürede ve daha az eylemle verilen görevi tamamlamışlardır.

Katılımcıların her bir ana hamle için harcamış oldukları sürenin, görev tamamlama süresindeki yüzdelik dağılımı ise Şekil 4.24'teki gibidir. Bu grafik dağılıma göre, tüm katılımcıların, modelleme süreçlerinin ilk %10'luk dilimlerinde daha hızlı kararlar aldıkları ve daha fazla hamle gerçekleştirdikleri; son %10'luk dilimlerinde ise nispeten daha az hamle gerçekleştirdikleri görülmektedir. Bu kapsamda, katılımcıların davranışlarında karar alma-uygulama yönünden herhangi bir örüntü benzerliğinden bahsetmek mümkün olmasa da gerçekleştirilen hamleler ve uygulama sıklıklarına bakıldığında; sürecin ilk yarısında katılımcıların daha hızlı strateji geliştirdiklerini söylemek mümkündür.



Şekil 4.24 : Katılımcıların uygulama süresince her bir ana hamle için harcadıkları sürenin uygulama sürecindeki yüzdelik dağılımı.



5. SONUÇ VE ÖNERİLER

Hesaplamalı Düşünme, eğitimde teknolojinin gelişmesine, öğrenim deneyimlerinin iyileştirilmesine katkıda bulunan ve tüm disiplinler içinde teşvik edilmesi gereken çok yönlü bir düşünme biçimidir. Bu nedenle, tasarım bilişim araçlarının yaygınlaşan kullanımıyla yeni bir paradigmaya yönelen mimarlık da ilk yıllardan itibaren eğitim müfredatında bu düşünme biçimine yer vermelidir. Nitekim, bilgisayarın tasarımda kullanımı geleneksel tasarım bilişinin ilk araştırma çerçevelerini nasıl tanımladıysa, hesaplamalı düşünmenin tasarımda kullanımı da hesaplamalı tasarım araştırmalarına karşı benzer bir meydan okuma halindedir. Bu tür araçlarla çalışmak, bir yandan tasarımcı kimliğiyle görsel, algısal ve uzamsal becerilere hâkim olmayı gerektirirken; diğer yandan, hesaplamalı işlemler hakkında fikir sahibi olmayı gerektirir. İşte bu noktada HD, temel bilişsel değerlerinin kullanımı çerçevesinde tasarımcıya eğitilmiş bir göz vaat eder. Dolayısıyla, mimarlık öğrencilerinin soyutlama, örüntü tanıma, ayrıştırma ve algoritmik düşünme becerilerini, tasarımcı bakış açısıyla ve farklı soyutlama biçimleri ile pratik yaparak geliştirmeleri oldukça önemlidir.

Bu araştırmada da HD'nin bilişsel değerlerine tasarımcı bakış açısıyla yer verilmiş ve bu bilişsel değerlerin mimarlıktaki mevcut öğretim kültürüne ışık tutulmuş; farklı düzeylerde soyutlamalar ve genellemeler yapabilmenin, görsel hesaplama sürecinde belirleyici rol oynadığı gösterilmiştir. Bu kapsamda, mimarlık eğitim araştırmalarında çoğunlukla göz ardı edilen prosedürel soyutlamanın HD üzerindeki belirleyici rolü ortaya çıkarılmış; tasarımcı bakış açısıyla hesaplamalı düşünmede prosedürel soyutlamanın değerlendirilmesi için bir yöntem önerilmiştir. Araştırmada, mesleki eğitimi süresince mühendislik bilimlerine göre farklılaşan kavramsal yaklaşım ve öğrenim deneyimleriyle soyutlama becerisi kazanmış olan mimarlık öğrencilerinin görsel hesaplama süreçlerini sistematik olarak analiz etmek ve geliştirilen değerlendirme yöntemini uygulamak üzere bir deney gerçekleştirilmiştir. Deneyde, görsel düşünür olarak nitelendirilmiş, ancak farklı tasarım uzmanlığına sahip katılımcıların bilişsel süreçleri, görme ve davranış biçimleri yönünden değerlendirmeye alınmıştır.

Geliştirilen değerlendirme yöntemi, kavramsal soyutlamaya yatkınlık gösteren görsel düşünürlerin prosedürel soyutlama yapmada zorlanabildiklerini ve bu durumun, görsel hesaplama süreçlerini olumsuz etkilediğini ispatlamıştır. Bu bağlamda araştırma, bilişim araçlarının kullanımı çerçevesinde soyutlamanın kavramsal yönüyle değerlendirilmesinin, HD'nin verimli kullanımı konusunda tek başına yeterli bir kriter olmadığını göstermiş; bu araçların kullanımındaki hesaplamalı işlemler nedeniyle, prosedürel soyutlamanın HD üzerinde belirleyici olduğunu ortaya koymuştur.

Ayrıca, katılımcıların CAD modelleme görevlerindeki performansları, HD'nin verimliliği ve etkinliğinin farklı soyutlama biçimleri arasında denge kurmak ya da başka bir deyişle, genellemek yapmakla yakından ilişkili olduğunu göstermiştir. Nitekim, Arnheim'in (2004) örneklediği gibi, "gerçek genelleme, bir bilim adamının kavramlarını ve sanatçının imgelerini mükemmelleştirme yoludur. Bu oldukça mekanik olmayan bir prosedürdür." Dolayısıyla, ister hesaplamalı işlemler üzerinden prosedürel düzeyde yapılsın ister görsel imgeler üzerinden kavramsal düzeyde yapılsın, soyutlama yapmak her zaman genelleme yapmayı beraberinde getirmektedir. Görsel bir düşünür için genelleme yapmak, rastgele veya sonsuz sayıda örneği bir araya getirmek değil; benzerlikler ve farklılıklar arasındaki örüntülerin tanımlanarak ayrıştırılmasıyla ortaya çıkmaktadır. Bu nedenle, bireyin kavramsal ve prosedürel yönden soyutlama eğilimini tespit etmek ve ihtiyaç duyduğu seviyeye uygun öğrenim deneyimleriyle ona bilişsel beceri kazandırmak, eğitimcilerin dikkat etmesi gereken faktörlerdendir. Bu çerçevede, tasarım bilişim araçlarıyla çalışan mimarlık öğrencileri, eğitim süreçlerinde her iki düzeyde de soyutlama becerisi kazanacak şekilde teşvik edilmelidir.

Tez kapsamında yapılan araştırmanın bütününe ilişkin çıkarımları ise aşağıda maddeler halinde açıklamak mümkündür:

- Yeni yaklaşımıyla HD, bilişim teknolojilerini aktif olarak kullanan disiplinler tarafından eğitilmiş bir göz kazandırmaya yardımcı bir zihin aracı olarak teşvik edilmelidir; mimarlık da buna dahil olmalıdır.
- Bu nedenle mimarlık eğitiminde, tasarımcıyı bilişim teknolojilerini kullanma hususunda daha donanımlı hale getirmeyi hedefleyen ve tasarımcının bilişsel becerilerini destekleyen eğitim araçlarının ve yeni eğitim stratejilerinin geliştirilmesine ihtiyaç duyulmaktadır.

- Uygun araçlar ve uygulama yöntemleriyle teşvik edildiğinde, HD'yi problem tanımlamak veya çözümlmek için kullanılabilecek etkili bir düşünme biçimi olarak görsel düşünörlere kazandırmak mümkündür. Ancak bir eğitimci için kilit nokta, HD'nin temel bilişsel değlerlerinin görsel düşünörlere için nasıl teşvik edilmesi ve bu değler arasında anlamlı ilişkilerin nasıl kurulabileceğı yönünde öğrenim deneyimleri sunabilmektir.
- Mühendislik bilimlerine kıyasla, tasarım bilimlerinin HD'nin temel değleri için sunduğı kavramsal yaklaşım ve öğrenim deneyimlerindeki farklılıklar, bu kavramın mimarlık eğitimindeki öğretim költürü ve pedagojik uyumu çerçevesinde de dikkate alınmalıdır. Bu bağlamda “tasarımcı yaklaşımıyla hesaplamalı düşünme” üzerine bilişsel araştırmalar hem müfredatı iyileştirmeye hem de araç ve teknoloji gelişimine katkıda bulunabilecek eğitim stratejilerinin geliştirilmesine önemli katkıda bulunacaktır.
- Tasarımda bilişim teknolojilerinin kullanımı çerçevesinde, görsel algısal ve uzamsal becerileri destekler nitelikte öğrenim deneyimlerine yer vermek oldukça önemlidir. Ancak bu deneyimlerle, bireyin soyutlama becerisini kavramsal yönden geliştirmek değil, aynı zamanda tasarım araçlarıyla çalışması için prosedürel yönden de geliştirmek hedeflenmelidir.
- Tasarım bilişim uygulamaları çerçevesinde, bireyin farklı çalışma ortamları ve araçları ile erkenden deneyim kazanması, arayüz kullanımında kendine özgü çözüm yöntemleri ve tasarım stratejileri geliştirebilmesi açısından önemlidir. Dolayısıyla, tasarımcı adayının prosedürel düzeyde soyutlama becerisini erkenden kazanması ve geliştirmesi için bilişim teknolojileri mimarlık eğitiminde ilk yarıyıldan itibaren teşvik edilebilir. Fakat bu çerçevede sunulacak eğitim uygulamaların içeriğinin oluşturulmasında, bireyin tasarım uzmanlığı dikkate alınmalıdır.
- Soyutlama sırasında yapılan genelleme, örüntü tanıyarak ve ayrıştırma yaparak bir problemi ele almada ve problemi formüle etmek için algoritma oluşturmada etkilidir. Bu nedenle soyutlama, görsel düşünörlere için de hesaplamalı düşünmenin en önemli bileşenidir.

- Görsel algısal becerilerin öne çıktığı mimarlık gibi tasarım alanlarında, öğrencilerin görsel hesaplama sürecinde kavramsal soyutlamaya yatkınlık gösterebileceği dikkate alınmalıdır. Nitekim, yapılan araştırma ile, farklı düzeylerdeki prosedürel ve kavramsal soyutlama becerilerine sahip bireylerin, görsel hesaplamada da farklı performanslar sergiledikleri görülmüştür.
- Bu doğrultuda, tasarım bilişim araçlarıyla çalışmak için eğitilmiş bir göze sahip olmanın hem kavramsal hem de prosedürel düzeyde soyutlama yapabilmekle ve bu soyutlamalar arasında dengeli bir ilişki kurabilmekle mümkün olduğu görülmektedir.

Son olarak tezin, mimarlık eğitiminde bilişim araçlarının öğretim kültürü çerçevesinde, bireye farklı düzeylerde soyutlama becerisi kazandıran ve bunu denetleyen bir öğrenim mekanizmasının geliştirilmesi için nitelikli bir kaynak oluşturduğunu söylemek mümkündür. Önerilen uygulama ve değerlendirme yöntemi, acemi tasarımcıların bilişim teknolojilerini kullanmada ihtiyaç duydukları kavramsal ve prosedürel soyutlama becerilerini test etmek ve geliştirmek üzere kurgulanmış bir eğitim aracına dönüştürülmesi için oldukça tanımlı bir çerçeve oluşturmuştur. Nitekim, yakın gelecekte araştırmada gerçekleştirilen deneyin, uygulama içeriği yönünden zenginleştirilerek daha fazla katılımcı ile yeniden gerçekleştirilmesi planlanmakta ve sunulan değerlendirme yönteminin geliştirilmesiyle birlikte bir dijital eğitim aracına dönüştürülmesi planlanmaktadır.

İlerleyen süreçte, deney uygulamasının iyileştirilmek üzere bir oyun motoru ile tasarlanması ve içerik yönünden zenginleştirilmesi hedeflenmektedir. Yine bu iyileştirmeye bağlı olarak, kullanılan değerlendirme yönteminin söz konusu uygulamanın bir parçası olarak geliştirilmesi planlanmaktadır. Bu araçla, acemi tasarımcıların soyutlamada zayıf ve kuvvetli oldukları yönleri tespit etmek ve buna bağlı olarak görsel hesaplama becerilerini geliştirmek için ihtiyaç duydukları özelleşmiş öğrenim deneyimini kazandırmak amaçlanmaktadır. Özellikle tasarım eğitiminde bu türde bir uygulamanın hem eğitimcilere hem de öğrencilere yapıcı geri bildirim sağlayacağı düşünülmektedir. Bilişim teknolojilerini tasarım eğitiminin ilk yıllarına entegre etmenin veya eğitim çabalarını güçlendirmenin daha iyi yollarını arayan eğitimciler için, bu türde bir eğitim aracının oldukça verimli olacağı düşünülmektedir.

KAYNAKLAR

- Aish, R., & Hanna, S.** (2017). Comparative evaluation of parametric design systems for teaching design computation. *Design Studies*, 52, 144-172.
- Arnheim, R.** (2004). *Art and visual perception: A psychology of the creative eye, fiftieth anniversary printing* (Rev. ed.). Berkeley and Los Angeles.
- Automation.** (n.d.). Merriam-Webster online sözlük. son erişim 20.01.2020. alındığı adres: <https://www.merriam-webster.com/dictionary/automation>
- Barr, V., & Stephenson, C.** (2011). Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community? *Acm Inroads*, 2(1), 48-54. <https://doi.org/10.1016/j.promfg.2015.07.761>
- Bhavnani, S. K., John, B. E., & Flemming, U.** (1999). The strategic use of CAD: An empirically inspired, theory-based course. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 183-190.
- Blackwell, A. F.** (2002). "What Is Programming?" In *14th Workshop of the Psychology of Programming Interest Group*. Brunel University.
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kamyli, P., & Punie, Y.** (2016). Developing Computational Thinking: Approaches and Orientations in K-12 Education. In *Proceedings of the EdMedia 2016 Conference*. <https://doi.org/10.2791/792158>
- Celani, G., & Vaz, C. E. V.** (2012). CAD scripting and visual programming languages for implementing computational design concepts: A comparison from a pedagogical point of view. *International Journal of Architectural Computing*, 10(1), 121-137.
- Chester, I.** (2007). Teaching for CAD expertise. *International Journal of Technology and Design Education*, 17(1), 23-35.
- Combéfis, S., den Schrieck, V. V., & Nootens, A.** (2013). Growing Algorithmic Thinking Through Interactive Problems to Encourage Learning Programming. *Olympiads in Informatics*, 7.
- Cuny, J., Snyder, L., & Wing, J. M.** (2010). Demystifying computational thinking for non-computer scientists. *Unpublished Manuscript in Progress, Referenced in Http://Www. Cs. Cmu. Edu/~CompThink/Resources/TheLinkWing. Pdf.*
- Czerkawski, B. C., & Lyman, E. W.** (2015). Exploring issues about computational thinking in higher education. *TechTrends*, 59(2), 57-65.
- Çolakoğlu, B., & Yazar, T.** (2007). An innovative design education approach: Computational design teaching for architecture. *METU JFA*, 24(2), 159-168.

- Denning, P. J.** (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, 60(6), 33–39. <https://doi.org/10.1145/2998438>
- Derry, S. J.** (1990). Flexible cognitive tools for problem solving instruction. *Annual Meeting of the American Educational Research Association, Boston, MA*.
- Dorst, K.** (2015). *Frame innovation: Create new thinking by design*. MIT press.
- Doyle, S., & Senske, N.** (2016). *Between design and digital: bridging the gaps in architectural education*.
- Dror, I. E.** (2011). Brain Friendly Technology: what is it and why do we need it? *Technology Enhanced Learning and Cognition*, 1–4.
- Futschek, G., & Moschitz, J.** (2011). Learning algorithmic thinking with tangible objects eases transition to computer programming. *In International conference on informatics in schools: Situation, evolution, and perspective*. Springer, Berlin, Heidelberg, 155-164.
- Futschek, G.** (2006). Algorithmic thinking: the key for understanding computer science. *In International conference on informatics in secondary schools-evolution and perspectives*. Springer, Berlin, Heidelberg. 159-168.
- Goldschmidt, G.** (2011). Avoiding design fixation: transformation and abstraction in mapping from source to target. *The Journal of creative behavior*, 45(2), 92-100.
- Guzdial, M.** (2010). Does contextualized computing education help?. *ACM Inroads*, 1(4), 4-6.
- Guzdial, M.** (2008). Education Paving the way for computational thinking. *Communications of the ACM*, 51(8), 25. <https://doi.org/10.1145/1378704.1378713>
- Gül, L. F., Çağdaş, G., Çağlar, N., Gül, M., Sipahioğlu, I. R., & Balaban, Ö.** (2013). Türkiye’de Mimarlık Eğitimi ve Bilişim Teknolojileri. *Mimarlıkta Sayısal Tasarım Sempozyumu, İTÜ. LEACH*, 32–37.
- Ham, D. D. A.** (2015). *Playful calculation: tangible coding for visual calculation* (Doctoral dissertation, Massachusetts Institute of Technology).
- Hu, C.** (2011). Computational Thinking – What It Might Mean and What We Might Do About It. *Computer Science Education*, 223–227.
- Jonassen, D. H.** (1992). What are cognitive tools? In *Cognitive tools for learning*. Springer, 1–6.
- Johnson, B. R.** (2017). *Design Computing*. New York: Routledge, <https://doi.org/10.4324/9781315680057>
- Kalelioglu, F; Gülbahar, Y; Kukul, V.** (2016). A Framework for Computational Thinking Based on a Systematic Research Review - ProQuest Central - ProQuest. *Baltic Journal of Modern Computing; Riga*, 4(3), 583–596.

- Khabazi, Z.** (2012). Generative algorithms with grasshopper version 2.0. *AA on-line book*.
- Khan, J. V., Soute, I. A. C., De Angeli, A., Piccinno, A., & Bellucci, A.** (2017). 6Th International Symposium on End-User Development. In *IS-EUD 2017 6th international symposium on end-user development IS-EUD 2017*.
- Klahr, D., & Carver, S. M.** (1988). Cognitive objectives in a LOGO debugging curriculum: Instruction, learning, and transfer. *Cognitive Psychology*, 20(3), 362-404.
- Lawson, B.** (2006). *How designers think*. Routledge.
- Knight, T. W.** (1999). Shape grammars: five questions. *Environment and Planning B: planning and Design*, 26(4), 477-501.
- Knight, T., & Stiny, G.** (2015). Making grammars: From computing with shapes to computing with things. *Design Studies*, 41, 8-28.
- Ko, J., & Steinfeld, K.** (2018). *Geometric Computation: Foundations for Design*. Routledge.
- Kurland, D. M., & Pea, R. D.** (1985). Children's mental models of recursive LOGO programs. *Journal of Educational Computing Research*, 1(2), 235-243.
- Locke, J.** (1847). An essay concerning human understanding. Kay & Troutman.
- Moreno-Leon, J., Roman-Gonzalez, M., & Robles, G.** (2018). On computational thinking as a universal skill: A review of the latest research on this ability. *IEEE Global Engineering Education Conference, EDUCON, 2018-April*, 1684-1689. <https://doi.org/10.1109/EDUCON.2018.8363437>
- NAAB.** (2009). National Architectural Accrediting Board. <<https://www.naab.org/international/substantial-equivalency/>>, erişim tarihi 2.11.2019.
- Nardelli, E.** (2019). Do we really need computational thinking?. *Communications of the ACM*, 62(2), 32-35.
- Nilsen, E., Jong, H., Olson, J. S., Biolsi, K., Rueter, H., & Mutter, S.** (1993). The growth of software skill: A longitudinal look at learning & performance. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, 149-156.
- NRC.** (2010). National Research Council *Report of a Workshop on the Scope and Nature of Computational Thinking*. <https://doi.org/10.17226/12840>
- Oxman, R.** (2017). Thinking difference: Theories and models of parametric design thinking. *Design studies*, 52, 4-39.
- Özçınar, H.** (2017). Hesaplamalı Düşünme Araştırmalarının Bibliyometrik Analizi. *Eğitim Teknolojisi Kuram ve Uygulama*, 7(2), 149-171.
- Papert, S.** (1981). *Mindstorms: Children, Machines and Powerful Ideas*. Basic Books, Inc.
- Pea, R. D., & Kurland, D. M.** (1984). On the cognitive effects of learning computer programming. *New ideas in psychology*, 2(2), 137-168.

- Perlis, A. J.** (1962). *The Computer in the University. Computers and the World of the Future*. M. Greenberger. Cambridge, Massachusetts, The MIT Press.
- Reas, C., & McWilliams, C.** (2011). *Form+ Code: in design, art, and architecture*. Princeton Architecture Press.
- Reisberg, D.** (2015). *Cognition: exploring the science of the mind: sixth international student edition*. WW Norton & Company.
- Romero, M., Lepage, A. and Lille, B.** (2017) ‘Computational thinking development through creative programming in higher education’, *Int. J. Educ. Technol. High. Educ.*, 14(1). doi: 10.1186/s41239-017-0080-z.
- Rosson, M. B.** (1983). Patterns of experience in text editing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 171-175.
- Sadeghi, S., Dargon, T., Rivest, L., & Pernot, J. P.** (2016). *Capturing and analysing how designers use CAD software*.
- Schön, D. A.** (1987). *Educating the reflective practitioner*.
- Selby, C. and Woollard, J.** (2013). ‘Computational thinking: the developing definition’. erişim linki: <<https://eprints.soton.ac.uk/356481/>>
- Sengupta, P., & Kinnebrew, J. S.** (2013). *Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework*. 351–380. <https://doi.org/10.1007/s10639-012-9240-x>
- Senske, N.** (2014). *Digital minds, materials, and ethics: linking computational thinking and digital craft*, 841–850.
- Senske, N.** (2011). *A Curriculum for Integrating Computational Thinking*, 91–98.
- Simon, H. A., & Newell, A.** (1971). Human problem solving: The state of the theory in 1970. *American Psychologist*, 26 (2), 145–159. <https://doi.org/10.1037/h0030806>
- Smithwick, D. J.** (2016). *Physical design cognition: an analytical study of exploratory model making to inform creative robotic interaction* (Doctoral dissertation, Massachusetts Institute of Technology).
- Stiny, G.** (2006). *Shape: talking about seeing and doing*. MIT Press.
- Stiny, G., & Gips, J.** (1971). Shape grammars and the generative specification of painting and sculpture. In *IFIP Congress*, 2(3), 125-135.
- Terzidis, K.** (2006). *Algorithmic architecture*. Routledge.
- Topalli, D. and Cagiltay, N. E.** (2018) ‘Improving programming skills in engineering education through problem-based game projects with Scratch’, *Computers and Education*. doi: 10.1016/j.compedu.2018.01.011.
- Tovey, M.** (2015). Developments in design pedagogy. *DS 82: Proceedings of the 17th International Conference on Engineering and Product Design Education (E&PDE15), Great Expectations: Design Teaching, Research & Enterprise, Loughborough, UK, 03-04.09. 2015*, 188–193.
- Tversky, B.** (2011). Visualizing Thought: Topics in Cognitive Science(2010). *Topics in Cognitive Science*, 3(3), 499–535. <https://doi.org/10.1111/j.1756-8765.2010.01113.x>.

- TYYÇ. (2016).** TÜRKİYE YÜKSEKÖĞRETİM YETERLİLİKLER ÇERÇEVESİ YÖNETMELİĞİ, Lisans düzey yeterlilikleri, url: <http://www.ttyc.yok.gov.tr/?pid=33>, Son erişim 11.12.2018.
- Url-1** <<https://education.microsoft.com/Story/Course?token=4dFPG>>, son erişim 20.01.2020.
- Url-2** <<https://www.bbc.co.uk/bitesize/topics/z7tp34j>>, son erişim 20.01.2020.
- Url-3** <<https://developers.google.com/blockly>>, son erişim 20.01.2020.
- Url-4** <<https://el.media.mit.edu/logo-foundation/resources/onlogo/index.html>>, erişim tarihi 20.01.2020.
- Url-5** <https://docs.google.com/document/d/1i0wg-BMG3TdwsShAyH_0Z1xpFnpVcMypYJceHGWex_c/edit>, erişim tarihi 20.01.2020.
- Url-6** <<https://www.australiancurriculum.edu.au/f-10-curriculum/technologies//Glossary/?term=Computational+thinking#>>, erişim tarihi 20.01.2020.
- Url-7** <<https://www.barefootcomputing.org/concept-approaches/computational-thinking-concepts-and-approaches>>, erişim tarihi 20.01.2020.
- Url-8** <<http://www.sis.itu.edu.tr/tr/icerik/icerik.php?subj=MIM&numb=120E>>, erişim tarihi 20.01.2020.
- Url-9** <<https://www.tes.com/lessons/iq6Tg9j5xbOwqQ/abstraction-from-reality>>, erişim tarihi 20.01.2020.
- Url-10** <<http://www.drscratch.org/learn/Abstraction/>>, erişim tarihi 20.01.2020.
- Url-11** <<https://johnrausch.com/PuzzlingWorld/chap03.htm>>, erişim tarihi 20.01.2020.
- Ware, C. (2008).** *Visual thinking for design*. Morgan Kaufmann.
- Weisberg, R. W., & Reeves, L. M. (2013).** *Cognition: from memory to creativity*. John Wiley & Sons.
- Wing, J. M. (2017).** Computational thinking's influence on research and education for all. *Influenza del pensiero computazionale nella ricerca e nell'educazione per tutti. Italian Journal of Educational Technology*, 25(2), 7–14. <https://doi.org/10.17471/2499-4324/922>.
- Wing, J. M. (2008).** Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717–3725. <https://doi.org/10.1098/rsta.2008.0118>.
- Wing, J. M. (2006).** Computational Thinking. *Magazine Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>.
- Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015).** Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies*, 20(4), 715–728. <https://doi.org/10.1007/s10639-015-9412-6>.

Yazici, S. (2019). Rule-based rationalization of form: learning by computational making. *International Journal of Technology and Design Education*, 1-21.



EKLER

EK A: Pilot alıřmalar sonunda gerekleřtirilen uygulama anket soru ve cevapları

EK B: Final alıřmaları iin katılımcılara uygulanan ankette yer alan n-deęerlendirme soruları ve yanıtları

EK C: Tasarımcı ve tasarımcı olmayan katılımcıların eylemlerinin ierięinin grafik ifadesi





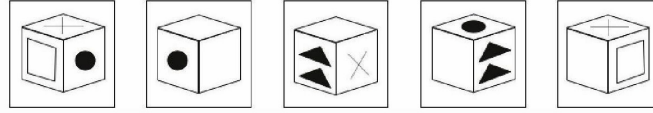
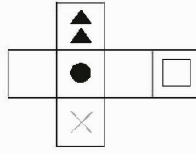
Çizelge A.1 : Pilot çalışmalar sonunda gerçekleştirilen uygulama anket soru ve cevapları.

Proje Dönemim	6	6	8	8	6	6	8	6	6	6	6	8	6	8	6
Öğrencilik Durumu	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim	Mimarlık öğrencisiyim
Daha önce Mimarlıkta Enformasyon Teknolojileri dersine benzer herhangi bir ders aldınız mı? Hayır/Evet ise ...	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır	hayır
Modelleme/kodlama/görselleştirme/çizim konusunda nereden ve hangi programlar için eğitim aldınız? [İtû içinden / seçmeli veya zorunlu ders]	Rhino	Photoshop	Diğer	SketchUp	Processing	Diğer	Diğer	3Ds Max	Rhino	Rhino, 3Ds Max	Diğer	SketchUp	3Ds Max	SketchUp, Photoshop	Diğer
Modelleme/kodlama/görselleştirme/çizim konusunda nereden ve hangi programlar için eğitim aldınız? [İtû dışından / kurs]	Autocad	Diğer	Diğer	SketchUp	Diğer	Diğer	Diğer	Diğer	Diğer	Diğer	Diğer	Diğer	Autocad, Rhino, SketchUp, Photoshop	Diğer	Autocad, Diğer
Modelleme/kodlama/görselleştirme/çizim konusunda nereden ve hangi programlar için eğitim aldınız? [Kendim öğrendim]	Revit	Autocad, SketchUp	Autocad, SketchUp, Photoshop	Autocad, Photoshop	Autocad, Rhino, Photoshop, Illustrator, Diğer	Autocad, Rhino, Photoshop	Autocad, Rhino, SketchUp, Photoshop, Illustrator	Autocad, Rhino, SketchUp, Photoshop, Illustrator	Autocad, SketchUp, Photoshop, Illustrator, Diğer	Autocad, SketchUp, Photoshop, Illustrator, Diğer	Autocad	Autocad, Rhino, Photoshop	Photoshop, Illustrator	Archicad, SketchUp, Photoshop, Illustrator	Revit, SketchUp, Photoshop, Illustrator, Diğer
Modelleme/kodlama/görselleştirme/çizim konusunda nereden ve hangi programlar için eğitim aldınız? [İşyeri/Staj/Diğer]	3Ds Max	Diğer	Diğer	Autocad	Autocad, SketchUp	Autocad	Revit	Diğer	Diğer	Revit	Diğer	Illustrator	Diğer	Autocad	Diğer
Projede kavramsal tasarım sürecinde aktif olarak kullandığım yazılım/programlar (Her satır için birden fazla seçenek işaretleyebilirsiniz.) [İki boyutlu çalışmalarında (çizimde) kullandığım yazılım/programlar]	Autocad	Autocad	Autocad	Autocad	Rhino, Photoshop, Illustrator	Autocad, Rhino, Photoshop, Illustrator	Autocad, Photoshop, Illustrator	Autocad	Autocad	Autocad, Photoshop, Illustrator	Autocad	Autocad, Photoshop	Autocad, Rhino	Autocad	Autocad, Revit
Projede kavramsal tasarım sürecinde aktif olarak kullandığım yazılım/programlar (Her satır için birden fazla seçenek işaretleyebilirsiniz.) [Üç boyutlu çalışmalarında (modellemede) kullandığım yazılım/programlar]	3Ds Max	SketchUp	SketchUp	SketchUp	Rhino, Illustrator	Rhino, Photoshop	Rhino	Rhino, SketchUp	Rhino, SketchUp	Rhino, SketchUp, Grasshopper/Dynamo	3Ds Max	Rhino, SketchUp, Grasshopper/Dynamo	Rhino, SketchUp	SketchUp	Revit, SketchUp
Projede kavramsal tasarım sürecinde aktif olarak kullandığım yazılım/programlar (Her satır için birden fazla seçenek işaretleyebilirsiniz.) [Görselleştirme ve temsil sürecinde aktif olarak kullandığım yazılım/programlar]	3Ds Max	Photoshop, Illustrator	Photoshop	Photoshop	Rhino, Illustrator	Photoshop, Illustrator	Rhino, Photoshop, Illustrator	Photoshop, Lumion	Photoshop, Lumion, Vray	Lumion, Vray	Photoshop	Photoshop, Vray	Photoshop, Lumion	Lumion, Vray	Revit, SketchUp
Kavramsal tasarım sürecinde iki boyutlu çalışmalarında ağırlıklı olarak bilgisayar programlarından yararlanmaktayım.	5	5	2	4	3	5	3	5	4	5	2	1	3	4	2
Kavramsal tasarım sürecinde üç boyutlu çalışmalarında ağırlıklı olarak bilgisayar programlarından yararlanmaktayım.	4	5	2	5	6	4	5	5	5	5	2	5	5	6	3
Mimarî Proje derslerinde bilgisayar ortamında kullandığım tasarım araçlarına (çizim ve modelleme) analog tasarım araçlarından (eskiz ve maket) daha fazla yer veremekteyim.	3	4	9	4	7	5	5	7	7	5	3	7	2	7	3
Tasarım sürecinde CAD programlarını zorluk çekmeden kullanıyorum.	7	5	3	6	7	3	6	4	6	5	4	7	7	6	5
Verilen uygulamaları zorluk yaşamadan tamamladığımı düşünüyorum.	4	4	2	4	5	5	3	4	5	5	5	5	5	4	5
İlk uygulamada (Patterns) hedef-sonuç ilişkisi bağlamında sıra ve kurallara dikkat ederek "Algoritmik Düşünme" becerimi kullandım.	4	4	2	4	3	3	3	3	3	5	4	5	5	4	5
İlk uygulamada (Patterns) iki boyutlu şekillerle "Soyutlama" becerimi kullandım.	3	4	4	4	3	2	4	3	3	5	2	5	5	5	5
İlk uygulamada (Patterns) iki boyutlu şekiller üzerinden "Örüntü tanıma ve parçalama" becerilerimi kullandım	3	4	4	3	5	4	3	4	4	4	4	5	5	5	5
İkinci uygulamada (Facets) hedef-sonuç ilişkisi bağlamında sıra ve kurallara dikkat ederek "Algoritmik Düşünme" becerimi kullandım.	3	4	4	2	3	4	3	3	4	5	3	5	5	5	5
İkinci uygulamada (Facets) üç boyutlu şekillerle "Soyutlama" becerimi kullandım.	3	3	4	2	5	3	3	3	4	5	2	3	5	4	5
İkinci uygulamada (Facets) üç boyutlu şekillerle "Örüntü oluşturma/çözümleme" becerimi kullandım.	3	4	4	3	5	3	2	3	4	5	4	5	5	5	5
Her iki çalışma için de oyunlaştırmanın seri düşünmeme ve pratik kazanmama yardımcı olduğunu düşünüyorum.	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Hayır	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet	Kesinlikle Evet
TeamViewer ile yaptığımız çalışmalarda memnun kaldığım uygulama	Parkur Uygulaması - rekabetçi ortam	Kule Tasarımı - işbirlikçi ortam	Parkur Uygulaması - rekabetçi ortam	Kule Tasarımı - işbirlikçi ortam	Parkur Uygulaması - rekabetçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Parkur Uygulaması - rekabetçi ortam	Kule Tasarımı - işbirlikçi ortam	Kule Tasarımı - işbirlikçi ortam	Parkur Uygulaması - rekabetçi ortam



EK B

When folded into a solid shape, which image can be seen? Press the key for your answer.



A

B

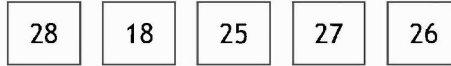
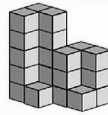
C

D

E

Tüm katılımcılar soruya 'E' olarak DOĞRU yanıt vermiştir.

How many blocks do you see in the given image? Enter the number for your answer.



A

B

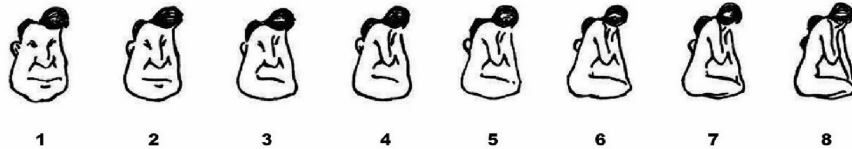
C

D

E

Tüm katılımcılar soruya 'E' olarak DOĞRU yanıt vermiştir.

At what numbers does the image change? Press the key for your answer.



1

2

3

4

5

6

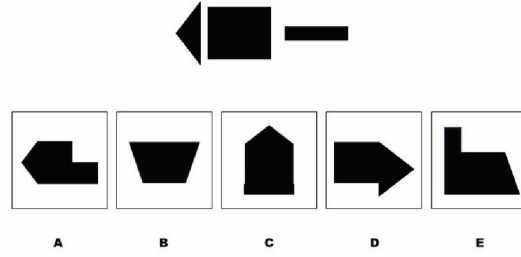
7

8

S-1 '5'; S-2 '4'; S-3 '6'; S-4 '5'; S-5 '5'; S-6 '5' olarak yanıtlamıştır.

Şekil B.1 : Final çalışmaları için katılımcılara uygulanan ankette yer alan ön-değerlendirme soruları ve yanıtları

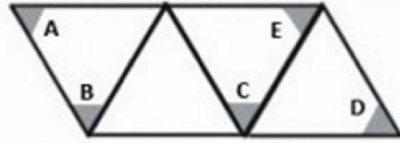
Which image can be made from the three shapes shown? Press the key for your answer.



Tüm katılımcılar soruya 'D' olarak DOĞRU yanıt vermiştir.

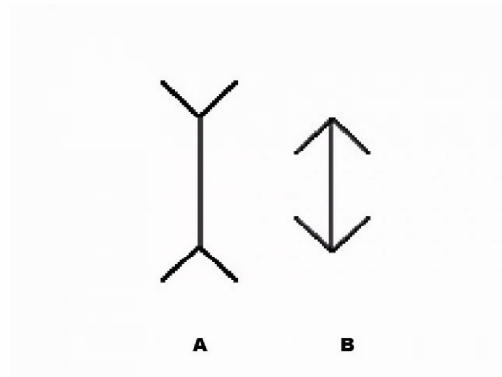
When folded into a solid shape, the corner "A" touches which other corner? Press the key for your answer.

C E B D



Tüm katılımcılar soruya 'E' olarak DOĞRU yanıt vermiştir.

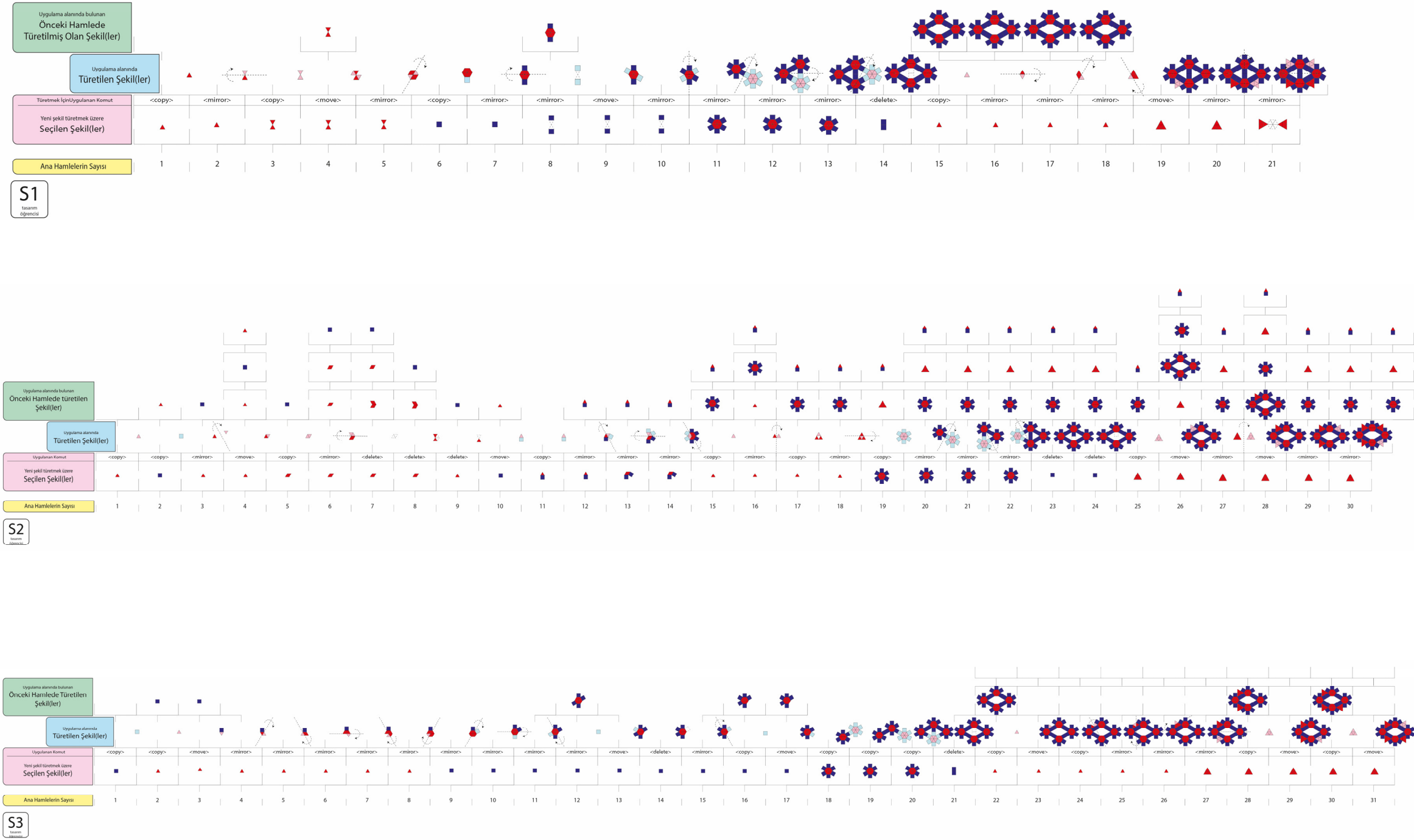
Which line appears to be longer? Press the key for your answer.



Tüm katılımcılar iki çizginin aynı boyda olduğunu DOĞRU tahmin etmiş; yanıtlarını sesli olarak söylemiştir.

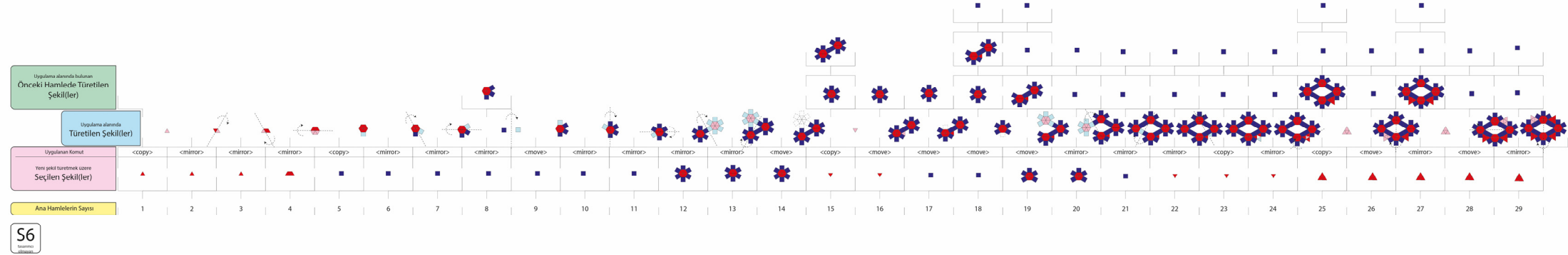
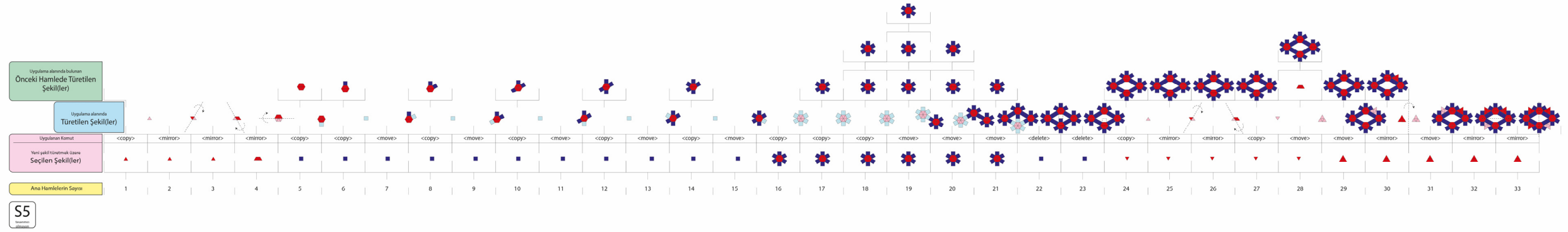
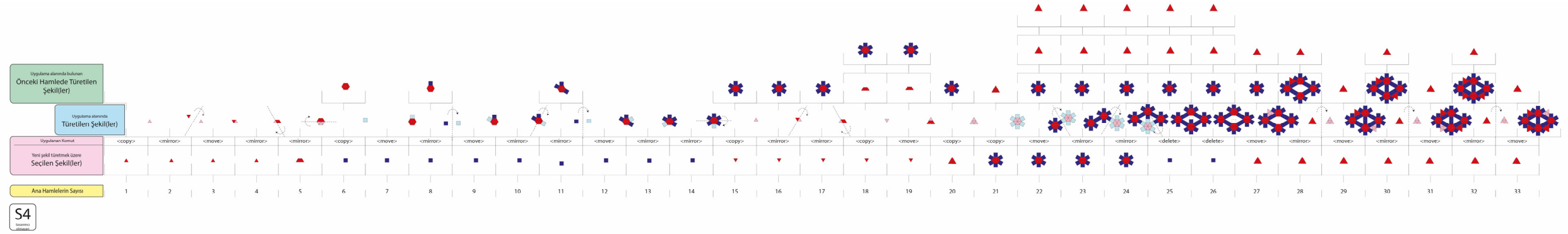
Şekil B.2 : Final çalışmaları için katılımcılara uygulanan ankette yer alan ön-değerlendirme soruları ve yanıtları.

EK C



Şekil C.1 : Tasarımcı katılımcıların eylemlerinin içeriğinin grafik ifadesi.





Şekil C.2 : Tasarımcı olmayan katılımcıların eylemlerinin içeriğinin grafik ifadesi.



ÖZGEÇMİŞ



Ad-Soyad : Elif Belkıs Öksüz
Doğum Tarihi ve Yeri : 1988 Kahramanmaraş
E-posta : elifb8807@gmail.com

ÖĞRENİM DURUMU:

- **Lisans** : 2011, Mimar Sinan Güzel Sanatlar Üniversitesi, Mimarlık Fakültesi, Mimarlık Bölümü
- **Yükseklisans** : 2013, İstanbul Teknik Üniversitesi, Mimarlık Anabilim Dalı, Mimari Tasarım Yüksek Lisans Programı

MESLEKİ DENEYİM VE ÖDÜLLER:

- 2017’de ERASMUS+ KA203 kapsamında hibe almaya hak kazanan TR01-KA203-046577 nolu PUDCAD - Practicing Universal Design Principles in Design Education through a CAD-based Game adlı Avrupa Birliği tarafından desteklenen Araştırma Projesi kapsamında, Araştırmacı olarak çalışmalarına devam etmektedir.
- 2012 – 2020 yılları arasında İstanbul Teknik Üniversitesi, Mimarlık Bölümü, Mimarlık Fakültesi’nde Araştırma Görevlisi olarak çalıştı.
- 2014 İstanbul Tasarım Bienali – Makale Yarışması’nda Teşvik Ödülü kazandı.

DOKTORA TEZİNDEN TÜRETİLEN YAYINLAR, SUNUMLAR:

- **Öksüz, E.B., Çağdaş, G.**, “An Assessment Method for a Designerly Way of Abstraction in Computational Thinking” ITU J Faculty Arch. (24.03.2020’de kabul almıştır; doi numarası için beklenmektedir).
- **Öksüz, E.B., Çağdaş, G.**, “The Convergence of Computational Thinking and Its Cognitive Aspects in Architectural Education”, International Journal of Advanced Research, cilt: 3, sayı:5, 2020, sf: 29-39.

- **Oksuz, E.B.**, Çağdaş G., 2018. A designerly way of computational thinking, Education, Design and Practice – Understanding skills in a Complex World, Stevens Institute, New York / New Jersey. 17-19 June 2019.
- **Oksuz, E.B.**, “HeartMade: An installation for Algorithmic Design Experience” BDS2017: Between Data and Senses: architecture, neuroscience, and the digital worlds, University of East London (UEL), London, UK. March 23-24th, 2017.
- **Oksuz, E.B.**, "Paramtericism for Urban Aesthetics" in A Al-Attili, A Karandinou & B Daley (eds), ASCAAD 2016: *The 8th International Conference of the Arab Society for Computer Aided Architectural Design: Parametricism vs. Materialism: Evolution of digital technologies for development*. Imperial House Publishers, London, UK, ISBN: 978-0-9955691-0-2

DİĞER YAYINLAR, SUNUMLAR VE PATENTLER:

- Cordan, Ö.; Yurdakul Toker, C.; **Oksuz, E.B.**, Dincay D.; Semizoğlu, S; Designing an inclusive navigation app for Taşkışla Building,"*E3_ Universal Design Practice Conference III _Design and Ergonomics. Designing for Learning of Learning Experience"* , Università degli Studi di Firenze, Florence, May 10, 2019.
- Cordan, Ö.; Dincay D.; Yurdakul Toker, C.; **Oksuz, E.B.**, Vayvay G, S. 2018. Engelsiz İç Mekandan Engelsiz Kent Kullanımına Doğru: Proje Pudcad, *Şehir & Toplum Dergisi*, Sayı 11, Marmara Belediyeler Birliği, 2018
- Cordan, Ö., Yurdakul Toker, C., Dincay D., **Oksuz, E.B.**, Vayvay G, S. Experiencing Universal Design Principles Within Pudcad Project: Local Workshops In Turkey, *International Journal Of Design Education*, 14 (4), <https://doi.org/10.18848/2325-128X/CGP/v14i04>
- Cordan, Ö.; Dincay D.; Yurdakul Toker, C.; **Oksuz, E.B.**, Vayvay G, S. Practicing Universal Design Principles, In High School Environment: Istanbul And Ankara, *Man, Space And Inclsuion Symposium*, Hochschule Ostwestfalen-Lippe - University Of Applied Sciences And Arts, Detmold, Germany, October 19, 2018
- **Öksüz, E.B.**; Sarı, T. From borders to boundaries: Istanbul Land Walls. *ITU J Faculty Arch.* 2018; 15(2): 51-59, DOI: 10.5505/itujfa.2018.54366
- **Öksüz, E. B.**, “An Alternative way of Creating the Crystal Image in Architecture: The architectonics of body and senses through digital” *DAKAM 3rd International ArchDesign Conference*, Istanbul, Turkey, June 17-18th, 2016.
- **Öksüz, E.B.**, Çağdaş G., 2014: ‘‘Mimarlıkta Alometri Üzerinden İlişkisel Biçimlenme,’’ 8. *Mimarlıkta Sayısal Tasarım Ulusal Sempozyumu*, İzmir Yüksek Teknoloji Enstitüsü, İzmir, Türkiye, 26–27 Haziran, 2014, p. 13–20. (ISBN:978-975-6590-06-5)

- **Öksüz, E.B., Çağdaş G.**, 2014: “Towards a Relational Complexity: Managing the Heterogeneous Complexity in Architecture,” *Nexus 2014: Relationship between Architecture and Mathematics PhD Day Presentations*, Middle East Technical University, Ankara, Turkey, 9–12 June, 2014, p.44—49. (ISBN:978-975429-329-6)
- **Öksüz, E.B.**, 2014: “Embedding Allometric Principles to Digital Morphogenesis”, *2nd Biodigital Architecture Conference 2014*, Universitat Internacional de Catalunya, Barcelona, Spain, 4–6 June, 2014, p. 266—279. (eISBN: 978-84-686-5307-5, pISBN: 978-84-686-5306-8)
- **Öksüz, E.B., Çağdaş, G.** 2014: “A Relational Morphogenetic Design Research through Networks and Allometry in Architecture,” *19th International Conference of the Association for Computer-Aided Architectural Design Research in Asia 2014*, Kyoto Institute of Technology, Kyoto, Japan, 14–16 May, 2014.
- **Öksüz, E.B.**, 2013: “Generating through Allometry in Architecture: A design Approach for Relational Morphogenesis,” *16th International Generative Art Conference*, University of Politecnico di Milano, Milan, Italy, 10–12 December, 2013, p. 27. (eISBN: 978-88-96610-25-1)
- **Öksüz, E.B.**, 2013: “Mimarlıkta Ağ Topolojileri ve Alometri Üzerinden Morfogenetik Tasarım,” MSc. Thesis in Architectural Design, Thesis Advisor: Çağdaş, G., Istanbul Technical University, Istanbul, Turkey, July, 2013.
- **Öksüz, E.B.**, 2013: “Sustainability of Biophilic Design,” *25th International Building & Life Fair and Congress*, Bursa Chamber of Architects, Bursa, Turkey, 28–30 March, 2013, p. 183—188. (pISBN: 978-605-01-0460-8)
- **Öksüz, E.B., Çağdaş, G.**, 2012: “Simulating the organization of multicellular organisms within spatial relations in architectural framework,” *15th International Generative Art Conference*, University of Politecnico di Milano, December 12–15, 2012, Lucca, Italy, p. 28. (eISBN: 978-88-96610-18-3)

Filename: İTÜ_Dr.Tezi_Siyah cilt teslimi_backup_backup
Directory: C:\Users\eeoks\Desktop
Template: C:\Users\eeoks\AppData\Roaming\Microsoft\Templates\Normal.dotm
Title:
Subject:
Author: İTÜ
Keywords:
Comments:
Creation Date: 5/8/2020 2:06:00 PM
Change Number: 3
Last Saved On: 5/8/2020 2:08:00 PM
Last Saved By: elifoksuz
Total Editing Time: 11 Minutes
Last Printed On: 5/8/2020 2:19:00 PM
As of Last Complete Printing
Number of Pages: 123
Number of Words: 26,333 (approx.)
Number of Characters: 150,103 (approx.)