



UĞUR BÖCEĞİM KODLAMA ROBOTU (U-Bot)

İÇİNDEKİLER

UĞUR BÖCEĞİM KODLAMA ROBOTU HANGİ AMAÇLA KULLANILIR?	3
TEKNİK ÖZELLİKLER.....	4
Şarj İşlemi	4
Alarm Modu	4
Kartlı Robotun Programlanması.....	4
Bakım	5
KARTLARIN GÖREVLERİ	6
SES VE BEKLEME KARTLARI.....	7
YÖN KARTLARI	7
HAZIR FONKSİYON KARTLARI	8
DÖNGÜ KARTLARI.....	9
HAFIZAYI TEMİZLEYEN KARTLAR	10
FONKSİYON OLUŞTURMA KARTI	10
PROGRAMLAMAYA AİT TEMEL KAVRAMLAR.....	11
KOD	11
ALGORİTMA.....	11
AKIŞ DİYAGRAMI	11
SIRALAMA	12
SIRALAMA ÖRNEKLERİ	13
HATA AYIKLAMA	14
HAYVANLAR ALEMİ ETKİNLİĞİ	15
ÖRNEK-1	16
ÖRNEK-2	17
ŞEKİLLER VE RAKAMLAR ETKİNLİĞİ	18
ÖRNEK-3 (DAİRE ÇİZİMİ 1)	19
SES SENSÖRÜ.....	20
DÖNGÜ KAVRAMI VE İÇ-İÇE DÖNGÜLER	21
ÖRNEK-4:.....	22

ÖRNEK-5:.....	23
ÖRNEK-6 (DAİRE ÇİZİMİ-2)	24
HAVA DURUMU ETKİNLİĞİ.....	25
ÖRNEK-7 (RÜZGAR GÜLÜ).....	26
ÖRNEK-8 (BALIK ÇİZİMİ)	27
ÖRNEK-9 (DAİRE ÇİZİMİ-3)	28
MÜZİK MATİ ETKİNLİĞİ.....	29
ÜLKELER MATİ ETKİNLİĞİ.....	30
FONKSİYON OLUŞTURMA	31
UYGULAMA YAPRAKLARI-1.....	33
UYGULAMA YAPRAKLARI-2.....	34
UYGULAMA YAPRAKLARI-3.....	35
YÖN OKLARI-1	36
YÖN OKLARI-2.....	37
YÖN OKLARI-3.....	38
YÖN OKLARI-4.....	39
GRAFİK DİKTE ÇALIŞMALARI.....	40
FİL (Örnek-1).....	41
Örnek-1.....	42
ANAHTAR (Örnek-2)	43
Örnek-2.....	44
BALIK (Örnek-3)	45
Örnek-3.....	46

UĞUR BÖCEĞİM KODLAMA ROBOTU HANGİ AMAÇLA KULLANILIR?

Robotumuz Okul Öncesi ve İlköğretim seviyesindeki Öğrencilerin eğitimde kullanılmak üzere tasarlanmıştır. Tasarlanan Robot ile Öğrenciler'e bilgisayardan bağımsız (ekransız) bir çalışma ile Kodlama Eğitimi'nin temellerini vermek amaçlanmıştır. Robotumuz manyetik kartlar ile kodlanmaktadır.

4-10 yaş grubuna hitap eden bu robot ile, bilgisayar programlarında kullanılan soyut 'Kodlar' manyetik kartlar kullanılarak somutlaştırılmaktadır. Böylece okuma yazma bilmeyen Okul Öncesi Öğrencileri bile rahatlıkla kodlama mantığını öğrenebileceklerdir.

Bu yöntemle çocuklara oyun destekli; Algoritma Oluşturma, Sıralama Yapma, Fonksiyon Oluşturma, Döngüler, Hata Ayıklama konuları öğretilenilecektir.

Robotun kullanımına yönelik hazırlanan kitapçığın, eğitim verecek olan Öğretmenler'e yardımcı olacağı düşünülmektedir.

Analitik düşünme ve odaklanma becerilerini geliştirecek bu eğitim ile çocukların bilişsel gelişimleri desteklenecektir.

Erken çocukluk döneminde kodlama eğitimi alan çocukların herhangi bir problemle karşılaştıklarında, yaşlarına göre daha kolay bir şekilde problemi analiz etme, işlem sırasını belirleme ve stratejik planlama yaparak, çözüme daha kolay ulaşabildikleri görülmüştür.

Şarj İşlemi

- Mikro USB soketine uyumlu herhangi bir adaptör ile şarj edilebilir.
- Alternatif olarak herhangi bir PC veya Laptop üzerinden de şarj edilebilir.
- Şarj olurken USB soketinde kırmızı bir ışık görülecektir. Pil tam şarj olduktan sonra ışık mavi veya yeşil renge dönecektir.
- Robot ancak bir yetişkin tarafından şarj edilmelidir.

Alarm Modu

- Robotumuz 5 dakika süresince kullanılmaz ise kullanıcıyı uyarmak için alarm verecek şekilde programlanmıştır.

Kartlı Robotun Programlanması

- Robotumuz manyetik kartlar ile programlanabilen bir zemin robotudur.
- Düğmesi açıldıktan sonra gözlerinin ışıkları yanıp sönecek ve robotumuz kodlamaya hazır hale gelecektir.
- Manyetik kartların okutulması sureti ile kodlanan robotumuz, “OYNAT” kartı okutulduktan sonra; kod dizisini adım adım işleyecektir.
- 22 adet farklı kod kartımız bulunmaktadır. Bunlar;

- | | |
|--------------|--|
| 1. Başla | 14. Melodi Çal |
| 2. Bitir | 15. Fonksiyon |
| 3. İleri | 16. Reset |
| 4. Geri | 17. Bekle (3sn.) |
| 5. Sağa Dön | 18. Geri Al |
| 6. Sola Dön | <u>Tekrar Kartları</u> |
| 7. 3X İleri | 19. İkili Tekrar Aç ve Kapat Kartları |
| 8. Sağ+İleri | " 2[" ve "] " |
| 9. Sol+İleri | 20. Üçlü Tekrar Aç ve Kapat Kartları |
| 10. 180° Dön | " 3[" ve "] " |
| 11. 360° Dön | 21. Sonsuz Tekrar Aç ve Kapat Kartları |
| 12. Ses | " ∞[" ve "] " |
| 13. Sallan | 22. Oynat Kartlarıdır. |

- Robotumuzun programlanması 'BAŞLA' kartının okutulması ile başlar ve 'BİTİR' kartının okutulması ile son bulur.

- En son 'OYNAT' kartının okutulması ile Robotumuz kod dizisini adım adım işleyecektir.
- Eğer kodlama sırasında yanlış bir algoritma oluşturulduysa; “RESET” kartı okutularak hafıza temizlenebilecektir.
- Robot İleri ve Geri yöne yaklaşık 15cm’lik adımlarla ilerlemektedir.
- Sağ ve Sol komutlarında ise 90⁰’lik açılarla dönmektedir.
- Robotumuz kod dizisini adım adım işleyip durduktan sonra; hafızasını temizlemektedir.

Bakım

- Robotunuzu direk güneş ışığından ve sıcaklıktan koruyunuz.
- Robotunuzu sudan uzak tutunuz.
- Robotunuzun üzerine herhangi bir baskı uygulamayınız. Aksi takdirde motorlarına zarar vermiş olursunuz.
- Robotunuzun hareketlerine itmek veya engel olmak suretiyle müdahale etmeyiniz. Aksi takdirde motorlarına zarar vermiş olursunuz.

KARTLARIN GÖREVLERİ

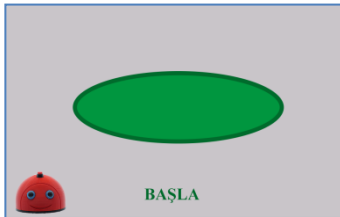
Sistemimizde programlama dillerindeki Kodları temsil eden Manyetik Kartlar kullanılmıştır. Soyut birer kavram olan kodlar, kartlar ile somutlaştırılarak Akış Diyagramı oluşturma temelli bir kullanım imkanı sağlanmıştır. Böylece okuma yazma bilmeyen okul öncesi öğrencileri dahil, tüm yaş gruplarına ekransız kodlama öğretme olanağı sunulmuştur. Manyetik kartların her biri farklı bir Kod'u temsil etmektedir. Yani her kartın belirli bir görevi vardır. Kartlar ön yüzleri Türkçe, arka yüzleri İngilizce ifadeler içerecek şekilde tasarlanmıştır. Benzer görevlerde olan kartlar aynı renkte tasarlanmıştır.

- Hareket Komutlarını içeren kartlar : Mavi
- Döngü Komutlarını içeren kartlar :Lacivert
- Hafıza Komutlarını içeren kartlar :Sarı
- Ses ve bekleme Komutlarını içeren kartlar : Lila
- Program Başla ve Bitir Kartları: Gri renkli olarak tasarlanmıştır.

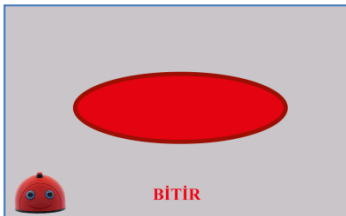
Bu kartlar belli bir sıra ile Robot'a okutulmakta ve alt alta dizilerek aynı zamanda programın akış diyagramı da görsel olarak elde edilmektedir.

Akış diyagramları Program'ın başladığını gösteren "BAŞLA" sembolü ile başlamakta ve işlemlerin bittiğini belirten "DUR" sembolü ile sona ermektedir. Bizim sistemimizde de algoritma oluştururken; Robotumuza okuttuğumuz ilk kart BAŞLA, son kart ise BİTİR kartı olmak zorundadır. Diğer kartlar ancak BAŞLA ve BİTİR kartlarının arasında okutulduğunda görevini yerine getirebilecek şekilde tasarlanmıştır. Aksi bir durumda Robotumuz tüm kartları okusa bile, OYNAT kartı okutulduktan sonra hata mesajı verecek ve programı çalıştırmayacaktır.

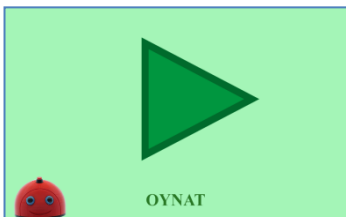
Temel kartlarımız aşağıda gösterilmiştir.



Bu kartın okutulması ile Robotumuz programlanmaya hazır hale gelir.

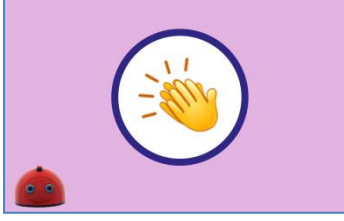


Bitir kartının okutulması ile, Robotumuz yazdığımız programın tamamlandığı anlar.



OYNAT kartının okutulması ile Robotumuz hafızasında yüklü olan programı adım adım çalıştırır.

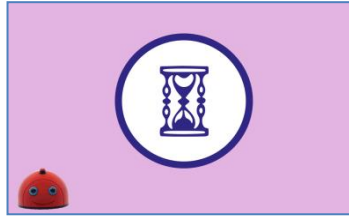
SES VE BEKLEME KARTLARI



Ses sensörünü aktive eden karttır. Pogramdaki işlem sırası bu komuta geldiğinde "El Çırpma" sesi duyulana kadar Robotumuz bekleyecek ve ses algılandıktan sonra, program bir sonraki komuta geçecektir.



Robotun müzik çalmasını sağlayan karttır.



Robotun 3 saniye boyunca beklemesini sağlar.

Bazı kartları işlevleri bakımından aynı başlık altında toplamak mümkündür. Bunlar aşağıda açıklanmıştır.

YÖN KARTLARI



Robotun İLERİ yönde 1 adım hareket etmesini sağlar.



Robotun GERİ yönde 1 adım hareket etmesini sağlar.



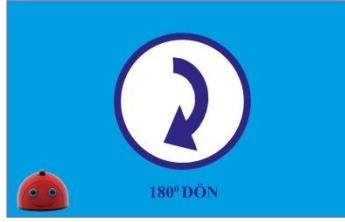
Robotun bulunduğu noktada, 90° lik bir açı ile SAĞ'a dönmesini sağlar.



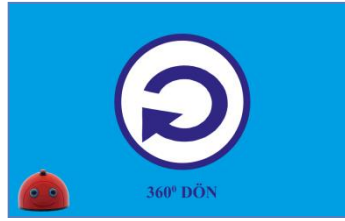
Robotun bulunduđu noktada, 90° lik bir açı ile SOL'a dönmesini sağlar.

HAZIR FONKSİYON KARTLARI

Birden fazla hareketin arka arkaya tanımlandığı ve hareketler arasında kesintisiz bir akış sağlayan hazır Fonksiyon kartlarıdır.



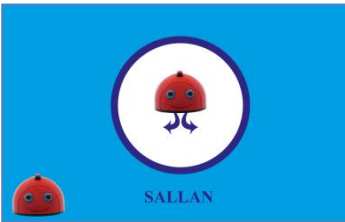
Robotun kendi etrafında 180° dönmesini sağlayan karttır.



Robotun kendi etrafında 360° dönmesini sağlayan karttır.



Robotun duraklamadan 3 adım İLERİ yönde yol almasını sağlayan karttır.



Robotun sağa ve sola sallanmasını sağlayan karttır.

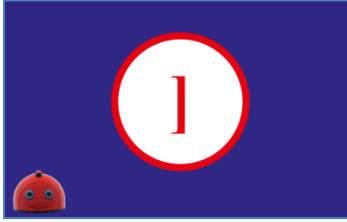


Robotun bulunduđu noktada SAĞ'a döndükten sonra, duraklamadan 1 adım İLERİ yönde yol almasını sağlayan karttır.



Robotun bulunduğu noktada SOL'a döndükten sonra, duraklamadan 1 adım İLERİ yönde yol almasını sağlayan karttır.

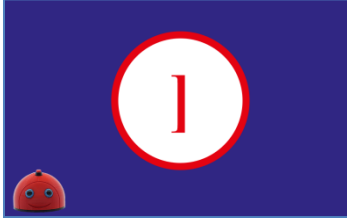
DÖNGÜ KARTLARI



Her iki kart beraber kullanılarak **İkili Döngü** oluşturulur.

- 1) Önce ilk kartı "**2[**" okutarak Döngümüzü açıyoruz.
- 2) Daha sonra istediğimiz kod kartlarımızı Robotumuza sırası ile okutuyoruz.
- 3) En sonunda ikinci kartımızı "**]**" okutuyor ve döngümüzü kapatıyoruz.

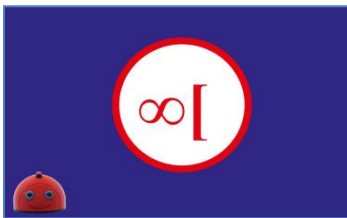
Böylece her iki kart arasında okutulan kodlar sırası ile **2 defa** tekrarlanacaktır.



Her iki kart beraber kullanılarak **Üçlü Döngü** oluşturulur.

- 4) Önce ilk kartı "**3[**" okutarak Döngümüzü açıyoruz.
- 5) Daha sonra istediğimiz kod kartlarımızı Robotumuza sırası ile okutuyoruz.
- 6) En sonunda ikinci kartımızı "**]**" okutuyor ve döngümüzü kapatıyoruz.

Böylece her iki kart arasında okutulan kodlar sırası ile **3 defa** tekrarlanacaktır.



Her iki kart beraber kullanılarak **Sonsuz Döngü** oluşturulur.

- 7) Önce ilk kartı "**∞[**" okutarak Döngümüzü açıyoruz.
- 8) Daha sonra istediğimiz kod kartlarımızı Robotumuza sırası ile okutuyoruz.
- 9) En sonunda ikinci kartımızı "**]**" okutuyor ve döngümüzü kapatıyoruz.

Böylece her iki kart arasında okutulan kodlar sırası ile **Sonsuz defa** tekrarlanacaktır. Sonsuz Döngü'den çıkmak için RESET kartının okutulması yeterlidir.

HAFIZAYI TEMİZLEYEN KARTLAR



Oluşturulan program akışında herhangi bir hata olduğu düşünülüyorsa RESET Kart'ı okutularak robotun tüm hafızası silinir.



Oluşturulan program akışında son okutulan komutu geri alan, yani silen karttır.

FONKSİYON OLUŞTURMA KARTI



Yeni bir Fonksiyon oluşturmak için kullanılır. Belli bir kod grubu ile oluşturduğumuz programı tek başına temsil eder. Programımızın gereksiz olarak uzamasını engeller.

KOD

Bilgisayarların anlayabileceği talimatlar **KOD** olarak adlandırılmaktadır. Kısacası KOD; bilgisayarlar ile iletişime girmemizi sağlayan kelimelerdir diyebiliriz.

Kodları kullanarak bilgisayarlara bir işlemi nasıl ve ne zaman yapabileceğini söyleriz.

Çok sayıda farklı programlama dili vardır. Hepsi farklı kelimeleri kod olarak kullanırlar. Bunu dünyada insanların konuştuğu farklı dillere benzetebiliriz . Örneğin; Türkçe "Merhaba" , İngilizce "Hello", Almanca "Hallo" gibi.

Programlama dilleri bilgisayarlara talimatlar verirken farklı kelimeler kullansalar da aynı işlemleri gerçekleştirmek için oluşturulmuşlardır.

ALGORİTMA

Bilgisayar biliminde algoritma, bir problemin çözümü için gerekli olan talimatların oluş sırasına göre, adım adım sıralanmasıdır.

AKIŞ DİYAGRAMI

Akış diyagramları algoritmaların, şekiller kullanılarak gösterildiği yapılardır.

Robotumuzu Programlarken kullandığımız kartların alt alta sıralanması ile, biz de kendi akış diyagramımızı elde ediyoruz.

SIRALAMA

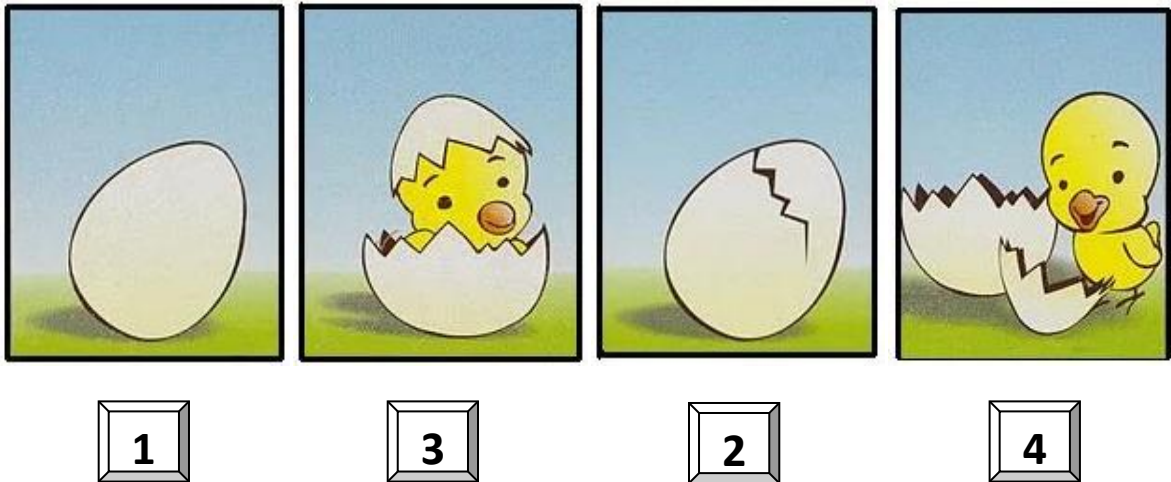
Programlama dillerinde komutlar; ilk komuttan son komuta kadar bir sıra ile işlenir. Birinci komuttan sonra ikinci komut, ikinci komuttan sonra üçüncü komut gibi. Bu işlem son komuta kadar devam eder. Bu sebeple doğru çalışan bir program yazmak istiyorsak, öncelikle doğru bir **Sıralama** yapmalıyız. Doğru bir sıralama ise ancak işlemlerin öncelikleri göz önünde bulundurulduğunda yapılabilir.

Robotumuzun programlanması kartlar ile yapıldığından, öğrencilerimiz kartları okuttuktan sonra alt alta sıraya dizecek ve programın görsel olarak **sıralaması** zaten oluşturulacaktır.

Sıralama konusunu öğrencilerimize anlatırken resimli örnekleri kullanmak ise çok faydalı olacaktır .

Sıralama Örneği:

Aşağıda resimleri oluş sırasına göre sıralayalım.



SIRALAMA ÖRNEKLERİ

Aşağıda resimleri oluş sırasına göre sıralayalım.

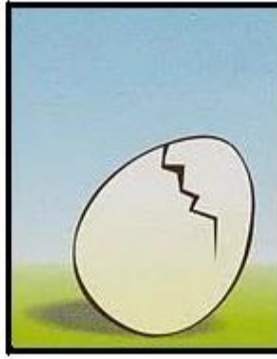
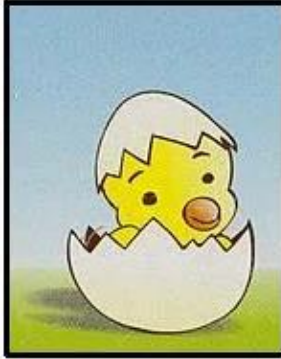
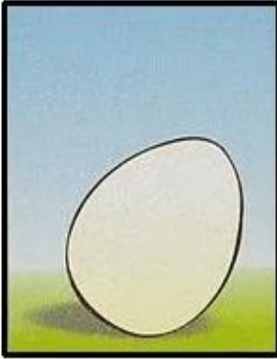
1)



2)



3)



HATA AYIKLAMA

Yazdığımız program sonucunda Robotumuz beklediğimizden farklı çalışıyorsa, ortada bir hata var demektir. O halde bu hatanın düzeltilmesi gerekmektedir. Programlama dillerindeki bu hataların düzeltilme sürecine **Hata Ayıklama** adı verilir.

Hataların oluşması birkaç farklı sebepten kaynaklanabilir. Bu sebeplerden bazılarını aşağıda sıralayabiliriz;

- Yapılması gereken işi analiz etmeden kod yazmaya başlamak.
- İş yanlış yorumlamak.
- Dikkatsiz davranarak farklı bir kod yazmak.

Programımızı oluşturduk ve Robotumuz farklı çalışıyorsa, hatamızı bulmak o kadar da zor değil. Çünkü programımızdaki kodları temsil eden kartlarımız ile oluşturduğumuz akış diyagramını takip ederek, hangi kartta hata yaptığımızı rahatlıkla tespit edebiliriz. Bu aşamadan sonra hatalı kartı sıralamamızdan çıkararak, yerine doğru kartı ekleriz ve yeni sıralamamızı Robotumuz'a tanıtarak doğru sonuca ulaşabiliriz.

HAYVANLAR ALEMİ ETKİNLİĞİ



Bu uygulamada öğrencilere kazandırılmak istenenler;

1. Hayvanların isimleri.
2. Hayvanların yaşadıkları yuvalar.

İhtiyaçlar:

1. U-Bot Kodlama Robotu,
2. Hayvanlar Alemi Matı.

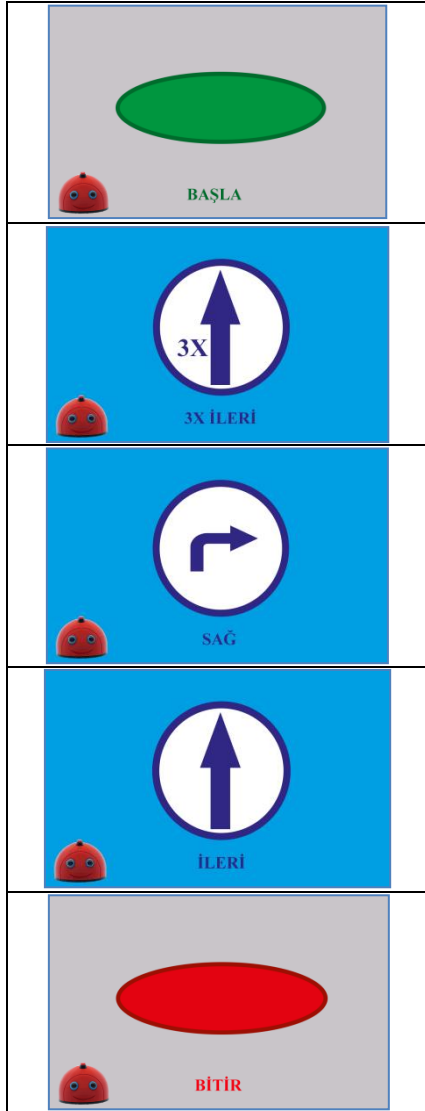
Uygulama:

1. Öğrencilerinizden Robotlarını herhangi bir hayvanın bulunduğu kareye yerleştirmelerini ve hayvanları yuvalarına götürecek şekilde kodlama yapmalarını isteyebilirsiniz.
 - Köpek-Kulübe
 - Arı-Kovan
 - Örümcek- Ağ
 - Penguen-İglo
2. Farklı bir uygulama olarak; Öğrencilerinize “Geyik, arkadaşı Köpeğin yanına gitmek istiyor ama yüzme bilmiyor, onun en uygun yoldan arkadaşının yanına gitmesini sağlayabilir misin?” şeklinde sorular yöneltebilirsiniz.

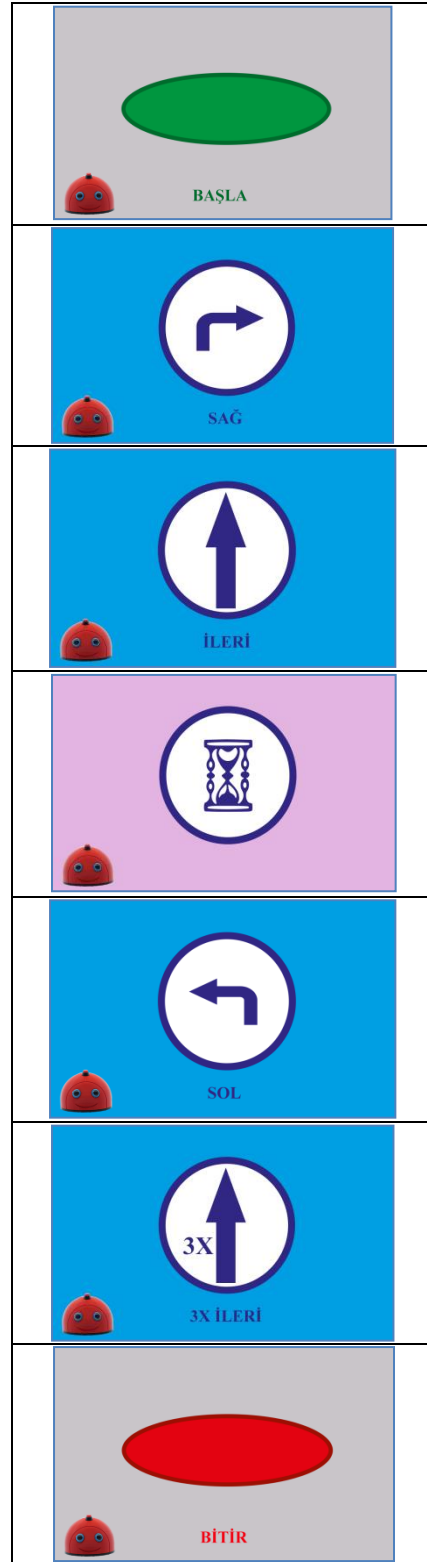
ÖRNEK-1

Örümceği yuvasına kaç farklı şekilde götürebilirsin

1) Robotumuz Örümcek karesinde ve yüzü Penguen'e dönük olsun. Örümceği yuvasına götüreceğ kodlamayı yapalım.



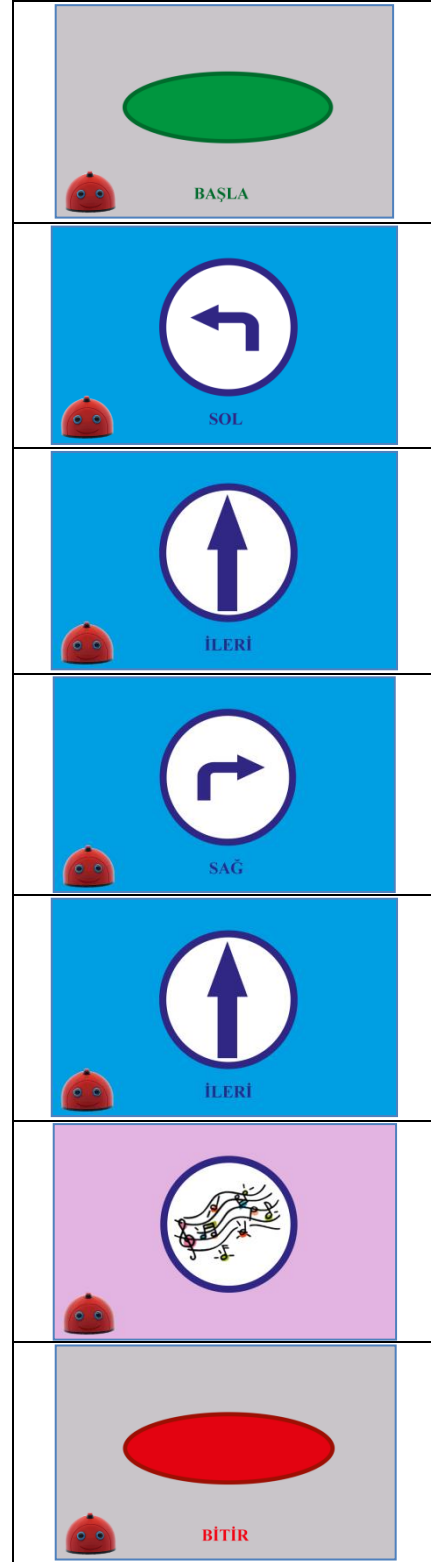
2) Robotumuz Örümcek karesinde ve yüzü Penguen'e dönük olsun. Robotumuz Ayı'ya uğrayacak, bir süre bu karede bekleyecek. Daha sonra ise Örümcek Yuvası'na gidecek kodlamayı yapalım.



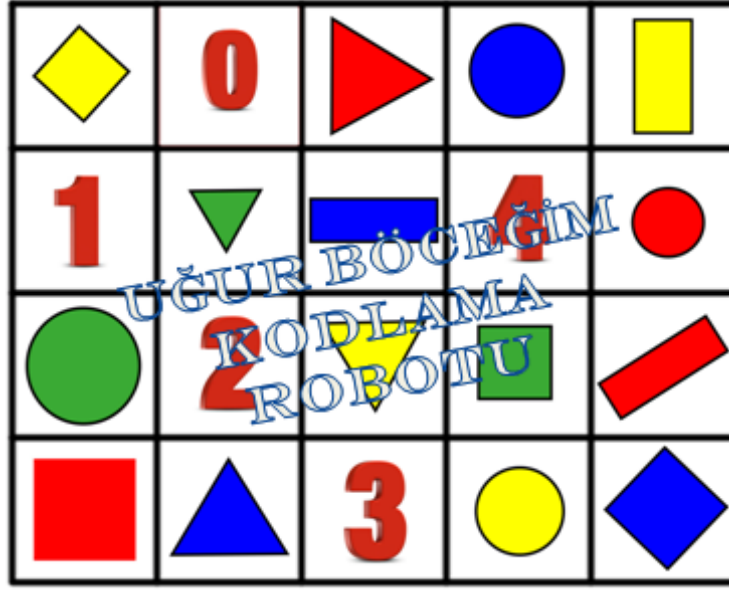
ÖRNEK-2

Robotumuz arkadaşı Geyik ile beraber denize bakıyorlar. Ama denizde çok büyük dalgalar var. Oysa arkadaşları Penguen'e gidip beraber müzik dinleyeceklerdi.

Onları en kısa yoldan arkadaşlarının yanına götürebilecek kodlamayı yapabilir misiniz?



ŞEKİLLER VE RAKAMLAR ETKİNLİĞİ



Bu uygulamada öğrencilere kazandırılmak istenenler;

1. Geometrik Şekiller ve özellikleri(Kenar ve Açı Sayısı)
2. Rakamlar ve Geometrik Şekiller ile ilişkileri.
3. Renk bilgileri.

İhtiyaçlar:

1. U-Bot Kodlama Robotu,
2. Şekiller ve Rakamlar Matı,
3. Şekiller, Renkler ve Rakamlara ait ders materyalleri.

Uygulama:

1. Mat üzerinde uygulamaya başlamadan önce; Öğrencilere Üçgen, Kare, Dikdörtgen ve Daire Şekilleri hakkında bilgiler verilir. Örneğin;
 - Kare ve Dikdörtgenin 4 adet kenar ve açıdan oluştuğu,
 - Üçgenin 3 adet kenar ve açıdan oluştuğu gibi.
2. İlk adımda verilen teorik bilgilerden sonra, mat üzerindeki uygulamalara geçilebilir.
3. Robotunuzu mat üzerindeki boş kareye yerleştirdikten sonra; Öğrencilerden robotu Mavi Üçgen'e götürecek kodlamayı yapmasını isteyebilirsiniz.
4. Başka bir uygulamada; Öğrencilerden Robotu 4 kenarlı bir şekle götürecek kodlamayı yapmasını isteyebilirsiniz.

ÖRNEK-3 (DAİRE ÇİZİMİ 1)

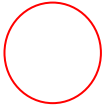
Şekiller ve Rakamlar Matı ile çalıştıktan sonra, öğrencilerinizle bir daire çizme uygulaması yapabilirsiniz. Örneğin: Bir balon çizebilirsiniz.

İhtiyaçlar:

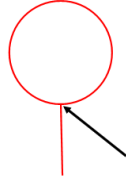
1. U-Bot Kodlama Robotu,
2. Kalem,
3. Bant.
4. Fon Kartonu

Uygulama:

1. Robotunuzun tam arkasına,bant yardımı ile bir kalem yapıştırınız.
2. Yandaki sütunda görülen kartları sırası ile robota okutunuz.
3. OYNAT kartını okuttuktan sonra robotunuz kod dizisini adım adım işleyecektir ve Şekil-1'deki gibi bir daire çizimi elde edeceksiniz.

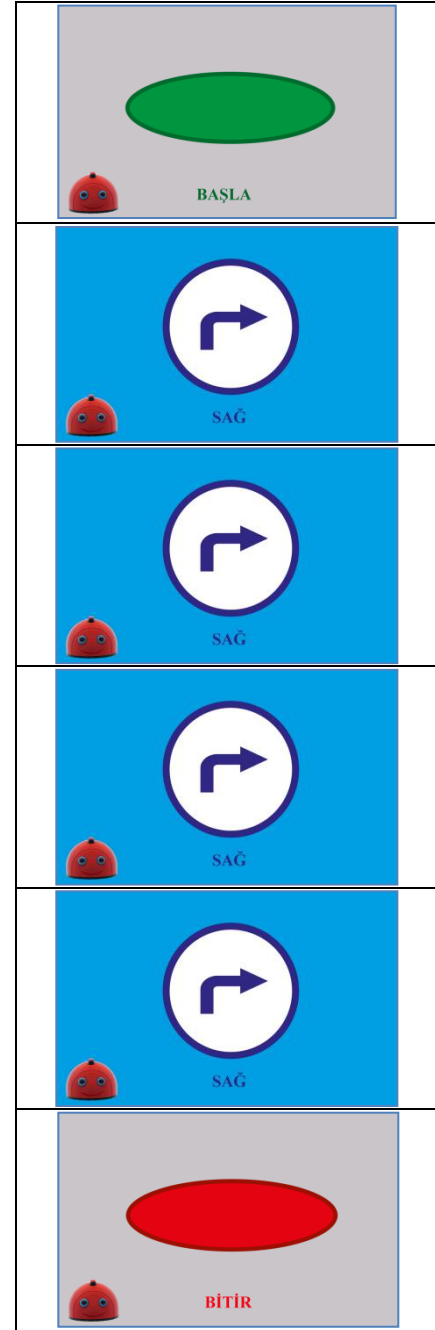


Şekil-1



Şekil-2

4. Balonun ipini çizmek için; kalemin ucu Şekil-2'de siyah ok ile gösterilen noktaya gelecek şekilde robotun konumunu değiştiriniz.
5. Robotunuzun 1 adım ileri gidecek şekilde hareket edebilmesi için



BAŞLA

İLERİ

BİTİR

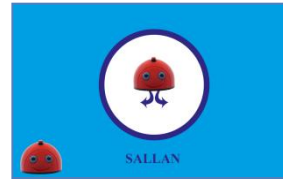
kartlarını okutunuz ve programınızı tamamlayınız.

6. OYNAT kartını okuttuktan sonra; Balonunuzun ipi çizilecektir.

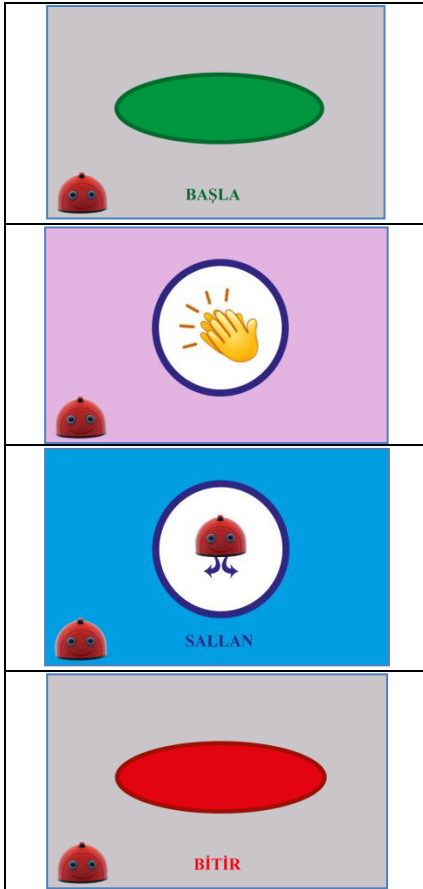
SES SENSÖRÜ

Canlılar çevresini duyu organları ile algılar. Robotlar ise sensörleri ile. Biz sesleri kulağımız ile duyarken, Robotlar Ses Sensörü ile çevrelerindeki sesleri algırlarlar. Zaten "Sensor" dediğimiz kelimenin Türkçe karşılığı "Algılayıcı"dır.

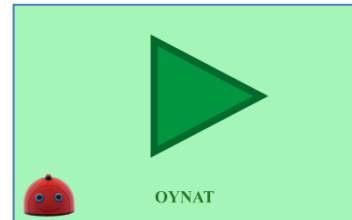
ÖRNEK: Robotunuzu ses duyduktan sonra sallanacak şekilde kodlayabilir misiniz?



CEVAP:



1. Kartlar yandaki sıraya göre Robot'a okutulurlar.



2. En son OYNAT kartı okutulur ve programın adım adım işlemeye başlar.

DÖNGÜ KAVRAMI VE İÇ-İÇE DÖNGÜLER

Döngü bir eylemin birden fazla defa tekrarlanmasıdır. Günlük hayatımızdan örnekler verecek olursak; dünyanın etrafında dönmesi, bisiklet sürerken pedal çevirmemiz, yürürken veya koşarken sürekli ayaklarımızın yer değiştirmesi birer döngüdür.

Programlama dillerinde de döngüler sıkça kullanılmaktadır. Programlama dillerinde istediğimiz Kod ya da Kod Dizilerini tekrarlamak için Döngüler kullanılır. Aynı kodları birden fazla yazmak yerine; Döngü yapılarının içine bir defa yazarak istediğimiz kadar tekrarlatırız. Böylece yazdığımız programı sadeleştirerek kısaltmış oluruz.

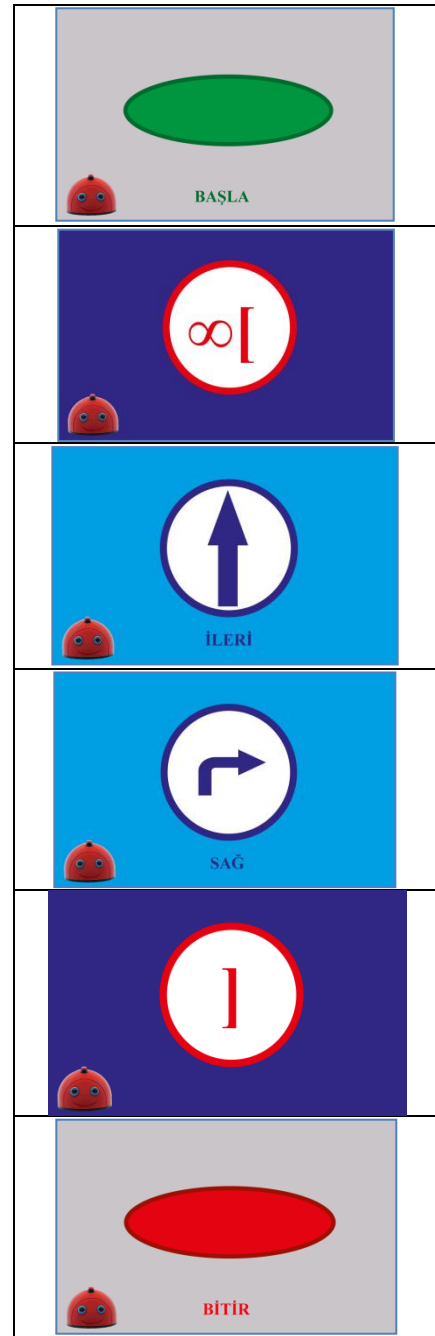
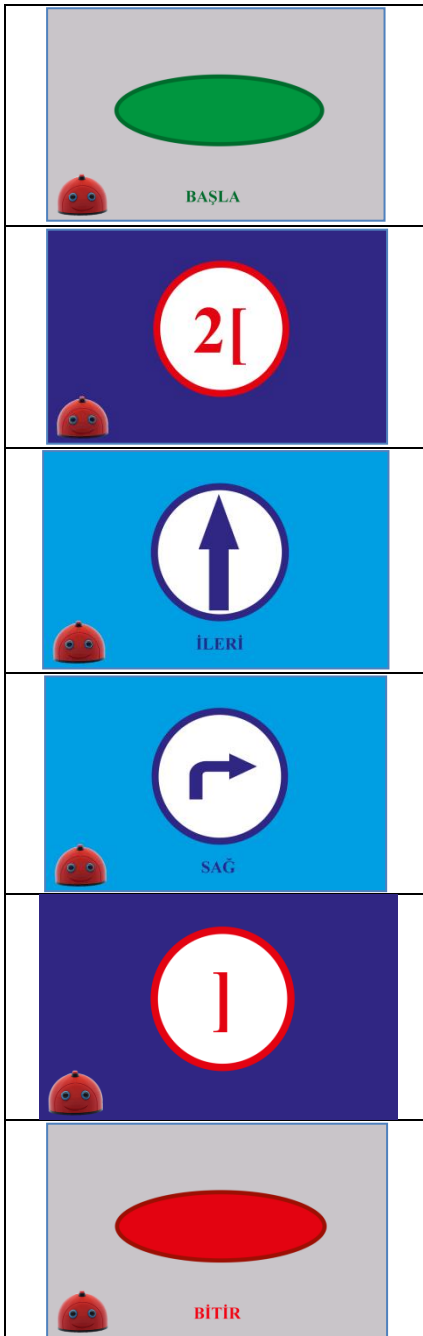
Bir döngü yapısının içerisinde, başka bir döngü yapısı kullanılmış ise ; elde edilen yeni yapıya **İç İçe Döngü** denir. Bu döngü yapısında dikkat etmemiz gereken nokta, komutlar işlenirken öncelikli olarak içteki döngünün tamamlanması daha sonra dıştaki döngüye geçilmesidir.

Döngüler işimizi kolaylaştırdığı için önemlidir. Robotumuzun 2, 3, ve ∞ defa döngüye girmesini sağlayacak kartlarımız mevcuttur. Robotumuz ile tekrar etmek istediğimiz komutları Döngü Aç ve Döngü Kapat Kartları arasında kullanmalıyız. Bu konuyu iki örnek ile açıklayalım.(ÖRNEK-4 ve ÖRNEK-5)

ÖRNEK-4:

Aşağıda İLERİ Git ve SAĞ'a Dön komutları iki farklı döngü yapısı içerisinde görülmektedir.

İlk örnekte komut dizimiz iki defa tekrarlandıktan sonra robotumuz duracaktır. İkinci örnekte, kod dizimiz sonsuz defa tekrarlanacak ve robotumuz sürekli İleri gidecek ve Sağa dönecektir. Sonsuz döngüye girmiş Robotumuzu durdurmak için ya kapatmalı ya da Reset Kartı'nı okutmalıyız.

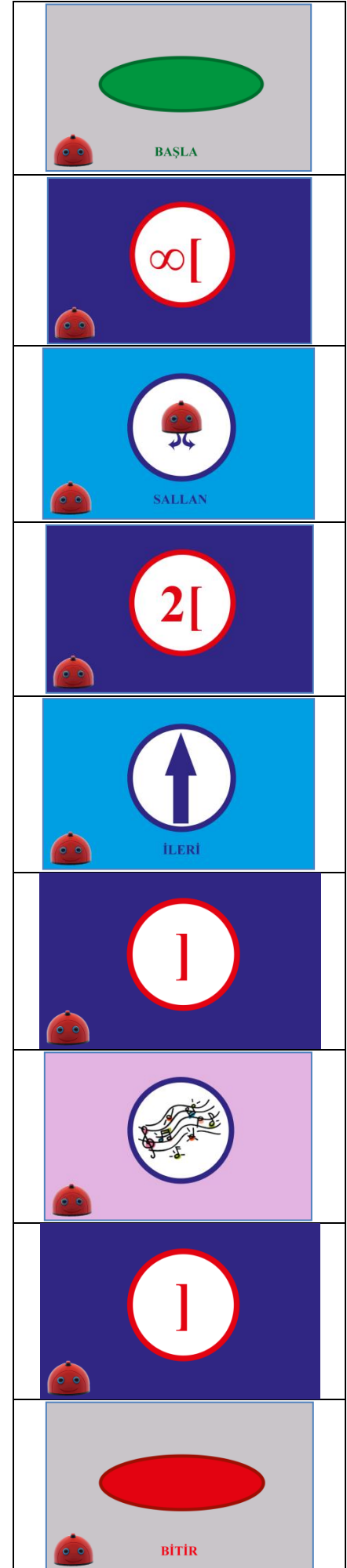


ÖRNEK-5:

Yandaki kod dizisinde İç-İçe Döngü yapısına bir örnek verilmiştir. Bu yapılarda en içteki döngü en önce tamamlanır.

Bu kod dizisini açıklayalım;

1. PLAY kartı okutulduktan sonra, programımız çalışmaya başlar ve sonsuz döngüye girer.
2. Robotumuz bir defa sallanır,
3. İki adım ileri gider ve İkili Döngü tamamlanır.
4. Bir defa müzik çalar ve sonsuz döngü tekrar başa döner.
5. Robotumuz tekrar sallanır,
6. İki adım ileri gider,
7. ve tekrar bir müzik çalar.
8. Bu durum sonsuz defa tekrarlanır.
9. Robotumuz biz enerjisini kesene kadar veya RESET kartını okutana kadar aynı işlemleri yapmaya devam eder.



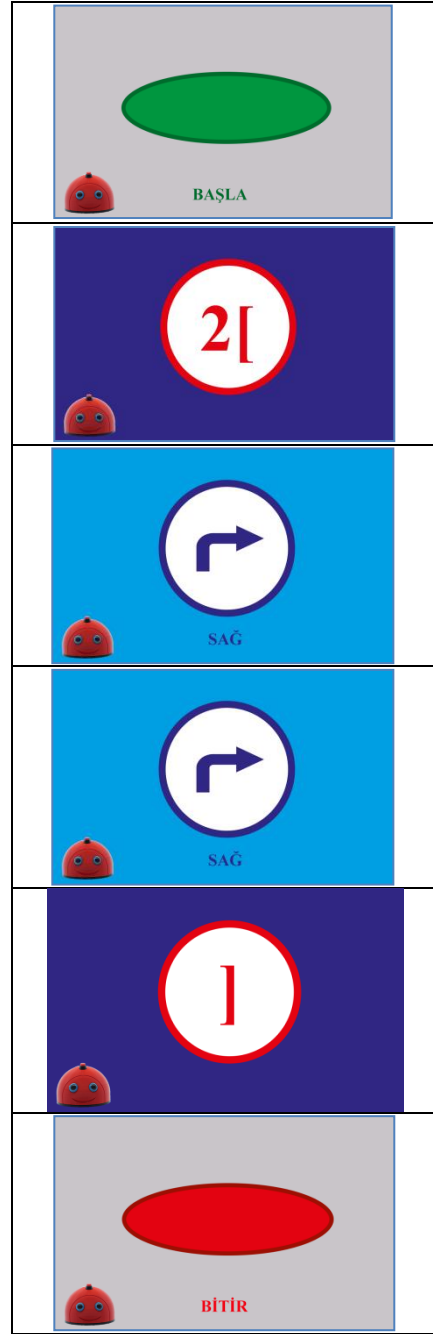
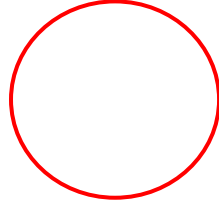
ÖRNEK-6 (DAİRE ÇİZİMİ-2)

Döngü yapısını kullanarak bir Daire çizmek için gerekli olan kodlamayı yapınız.

Uygulama:

1. Robotunuzun tam arkasına,bant yardımı ile bir kalem yapıştırınız.
2. Yandaki sütunda görülen kartları sırası ile robota okutunuz.
3. OYNAT kartını okuttuktan sonra robotunuz kod dizisini adım adım işleyecektir ve Şekil-1'deki gibi bir daire çizimi elde edeceksiniz.

Şekil-1



HAVA DURUMU ETKİNLİĞİ



Bu uygulamada öğrencilere kazandırılmak istenenler;

1. Mevsimler
2. Hava Durumları hakkında bilgiler.

İhtiyaçlar:

1. Kodlama Robotu,
2. Hava Durumu Matı,
3. Mevsimler ve Hava Durumlarını anlatan ders materyalleri.

Uygulama:

1. Kağıttan bir zar hazırlayıp, zarın her bir yüzeyine aşağıdaki şekilleri çiziniz;
 - Şimşek,
 - Yağmur Damlası,
 - Kar Tanesi,
 - Hortum,
 - Rüzgar Gülü
 - Gökkuşağı
2. Zarı atınız.
3. Örneğin zar yüzeyinde Yağmur Damlası denk geldi ise; Öğrencinizden Robot'u yağmur damlasının olduğu kareye yerleştirmesini ve Yağmur Bulutu'na götürecek şekilde kodlama yapmasını isteyebilirsiniz.

ÖRNEK-7 (RÜZGAR GÜLÜ)

'Hava Durumu' matı ile çalıştıktan sonra, öğrencilerinizle bir Rüzgar Gülü çizme etkinliği yapabilirsiniz.

Tekrarla kartlarını kullanmanız daha kısa bir program yazmanızı sağlayacaktır.

Bildiğiniz gibi ne kadar az kod kullanırsak, o kadar iyi çalışan bir program elde etmiş oluruz.

İhtiyaçlar:

1. Kodlama Robotu,
2. Kalem,
3. Bant.
4. Fon Kartonu

Uygulama:

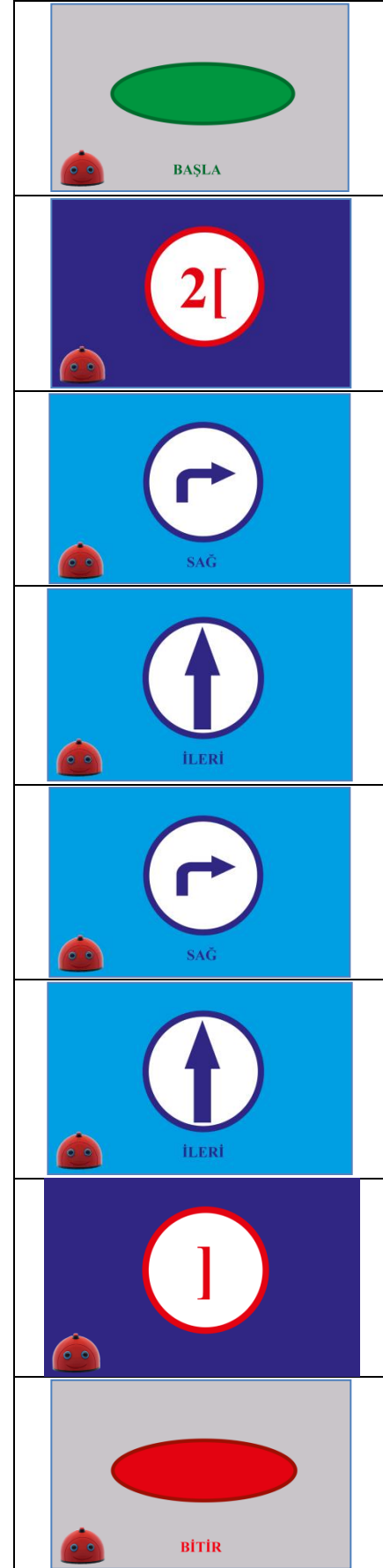
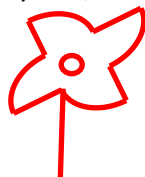
1. Robotunuzun tam arkasına, bant yardımı ile bir kalem yapıştırınız.
2. Daha sonra yan sütunda gördüğünüz kartları sırası ile Robotunuza okutunuz.
3. Bitir kartının okutulması ile program tamamlanmıştır.
4. OYNAT kartını okuttunuz. Robotunuz kod dizisini tamamladıktan sonra Şekil-1'deki çizimi elde edeceksiniz.

ŞEKİL-1



5. Elde ettiğiniz şekile bir sap ve merkezine bir daire ekleyerek, Rüzgar Gülü çiziminizi tamamlayabilirsiniz.

ŞEKİL-2



ÖRNEK-8 (BALIK ÇİZİMİ)

'Hayvanlar Alemi' matı ile çalıştıktan sonra, öğrencilerinizle bir Balık çizme etkinliği yapabilirsiniz.

İhtiyaçlar:

1. Kodlama Robotu,
2. Kalem,
3. Bant.
4. Fon Karton

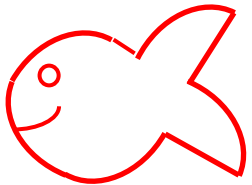
Uygulama:

1. Robotunuzun tam arkasına bant yardımı ile bir kalem yapıştırınız.
2. BAŞLA kartını okutunuz.
3. Yandaki sütunda görülen kartları sırası ile robota okutunuz.
4. BİTİR kartını okutunuz.
5. OYNAT kartını okuttuktan sonra robotunuz kod dizisini adım adım işleyecektir ve Şekil-1'deki gibi bir çizim elde edeceksiniz.

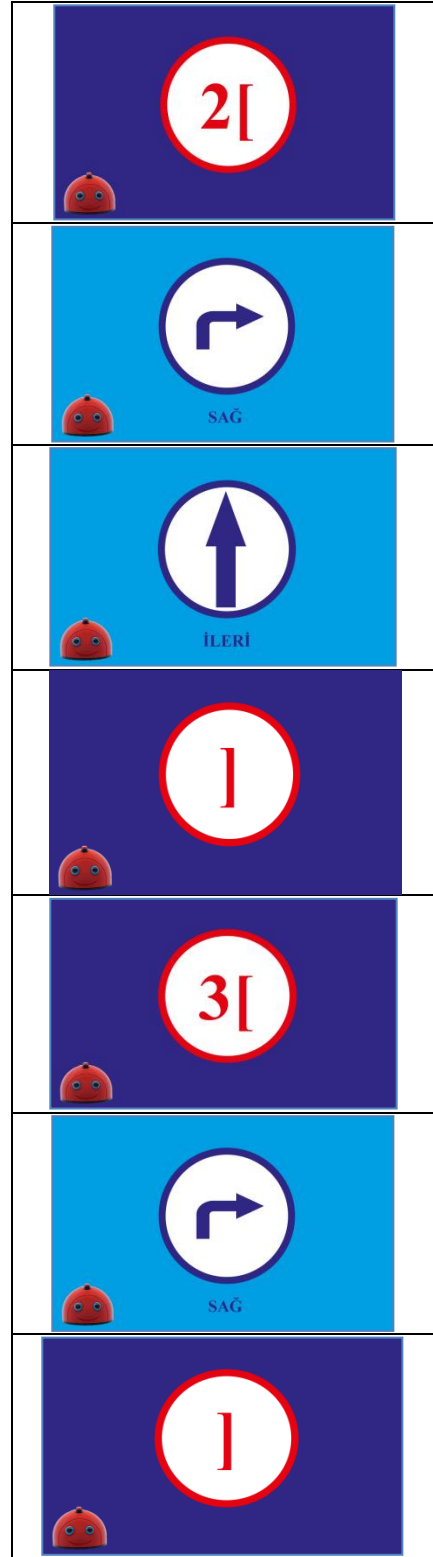


ŞEKİL-1

6. Çiziminize elinizle gerekli eklemeleri yaparak Şekil-2 'deki gibi Balık çiziminizi tamamlayınız ve isterseniz renklendiriniz.



ŞEKİL-2



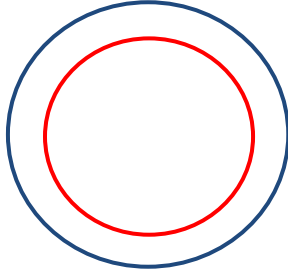
ÖRNEK-9 (DAİRE ÇİZİMİ-3)

Döngü yapısını kullanarak iç içe iki Daire çizmeye çalışınız.

Uygulama:

1. Robotunuzun tam arkasına ve yanına bant yardımı ile birer kalem yapıştırınız. Kalemlerin farklı renklerde olmasına özen gösteriniz.
2. Yandaki sütunda görülen kartları sırası ile robota okutunuz.
3. OYNAT kartını okuttuktan sonra robotunuz kod dizisini adım adım işleyecektir ve Şekil-1'deki gibi iç içe iki daireden oluşan bir çizim elde edeceksiniz.

Şekil-1



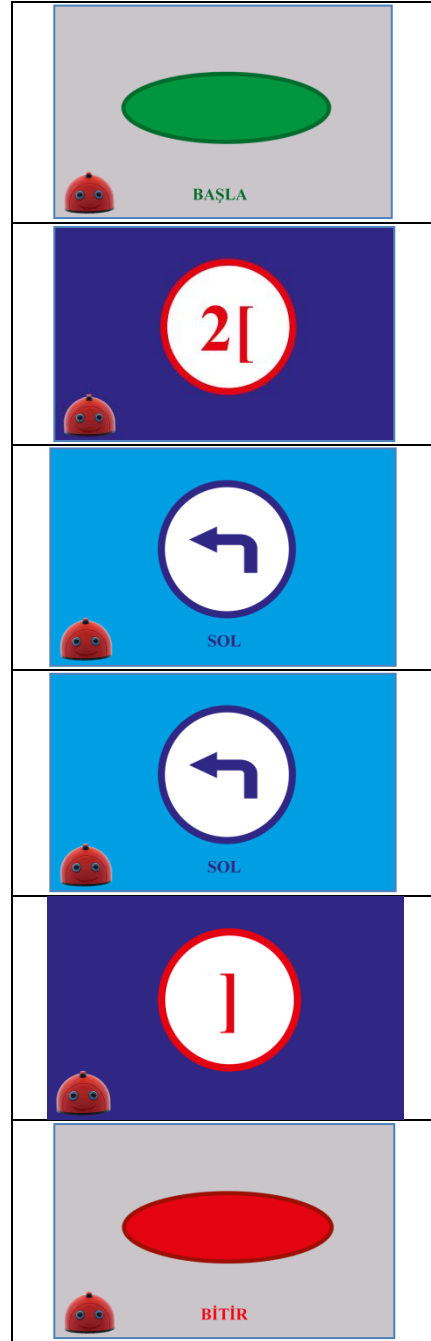
Bir Soru?

Neden Kırmızı Daire daha küçük, Mavi Daire daha büyüktür?

Cevap:

Kırmızı kalemi Robot'un merkezine daha yakın olan yan tarafına yapıştırdığımız için bu dairenin çapı daha küçüktür.

Mavi kalemi Robot'un merkezinden daha uzak bir noktaya yapıştırdığımız için, bu dairenin çapı daha büyüktür.



MÜZİK MATI ETKİNLİĞİ



İhtiyaçlar:

1. U-Bot Kodlama Robotu,
2. Müzik Matı,

Görev-1:

- U-Bot'u Gitar şeklinin bulunduğu karede, yüzü Kuyruklu Piyano'ya bakacak şekilde konumlandırınız.
- Kodlama sırasında zorunlu olarak kullanılacak kart; **180° DÖN**
- U-Bot'u Elektro Gitar'ın bulunduğu kareye ulaşacak şekilde kodlama yapınız.

Görev -2:

- U-Bot'u Gitarist Çocuğun bulunduğu karede yüzü mikrofona bakacak şekilde konumlandırınız.
- Kodlama sırasında **3X Döngü yapısı** zorunlu olarak kullanılacaktır.
- U-Bot'u Gitar'ın bulunduğu kareye ulaşacak şekilde kodlama yapınız.

Görev -3:

- U-Bot'u Def'in bulunduğu karede yüzü Elektro Gitar'a bakacak şekilde konumlandırınız.
- U-Bot'u tüm vurmali çalgıları gezecek ve bu karelere ulaştığında bekleyecek şekilde kodlayınız.

ÜLKELER MATI ETKİNLİĞİ



Bu uygulamada öğrencilere kazandırılmak istenenler;

Ülkelere ait bayrakları tanıyarak, önemli mekanları bulundukları ülke ile eşleştirmek.

İhtiyaçlar:

1. U-Bot Kodlama Robotu,
2. Ülkeler Matı,

Görev-1

- U-Bot Türkiye turuna çıkmak istiyor. U-Bot'u Türkiye'nin önemli yerlerinin bulunduğu karelerden geçecek şekilde kodlayınız.

Görev-2:

- U-Bot'u Pisa Kulesi'nin bulunduğu kareye konumlandırınız.
- U-Bot'u Mat üzerindeki diğer ünlü kulelerin üzerinden geçecek şekilde programlayınız.(Pisa Kulesi-Kız Kulesi-Eyfel Kulesi)

Görev-3:

- Bremen Şehri hangi ülkededir?
- Robotumuzu bu ülkenin bayrağının bulunduğu kareye konumlandırdıktan sonra, Bremen Mızıkacıları Anıtı'na ulaşacak şekilde kodlama yapınız.

Görev-4:

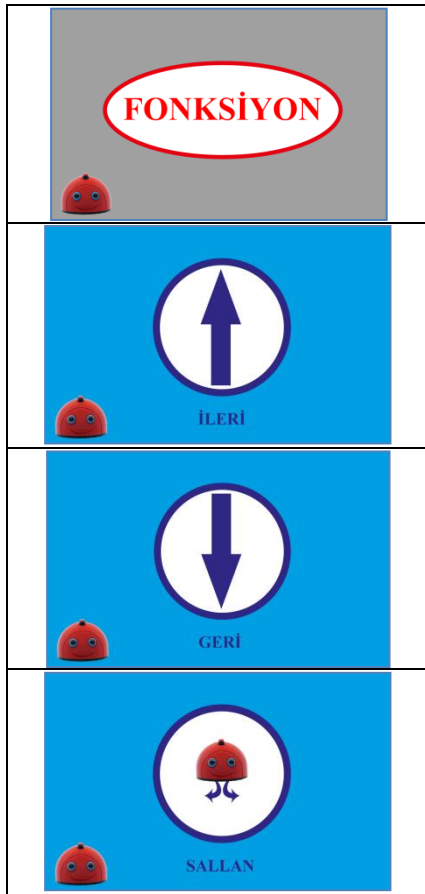
- Brüksel'in sembolü Atomium yapısını biliyor musunuz?
- Peki Brüksel Hangi ülkededir?
- U-Bot'u Belçika Bayrağı'ndan Atomium'a doğru kodlayalım.
- Kodlama sırasında **2X Döngü yapısı** zorunlu olarak kullanılacak.Unutmayalım!

FONKSİYON OLUŞTURMA

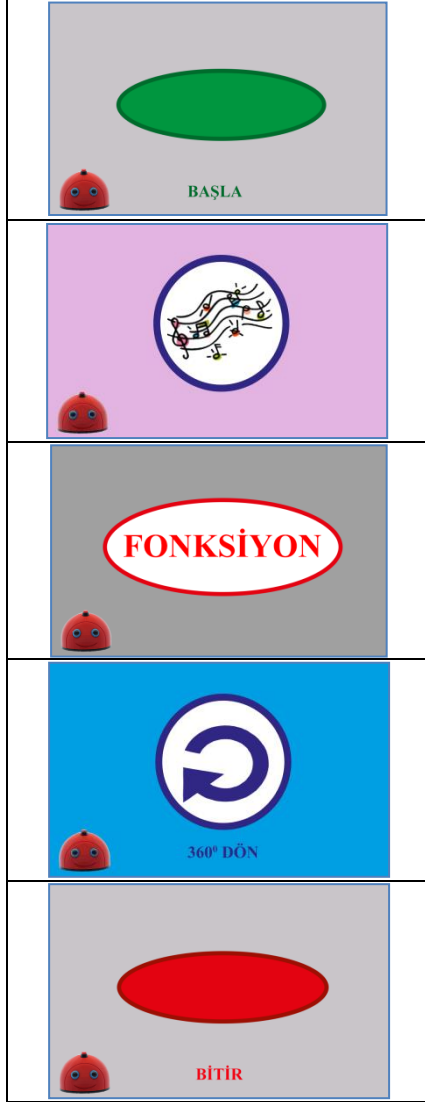
Fonksiyon kartını kullanarak kendinize ait fonksiyonlar oluşturabilirsiniz. Fonksiyonlar belli bir işlemi yapmak ile görevlendirilmiş yapılardır ve ana programınızın hangi noktasında çağırırsanız çalışacak kod dizileridir.

Bizim sistemimizde de Fonksiyon Kartı, program yazmadan önce Robotumuza tanıttığımız kod dizilerini, yazdığımız ana program içerisinde istediğimiz noktada **tek bir kart olarak temsil edecektir.** Bunu bir örnek ile açıklayalım. **Bu örnekte Robotumuz'a dans ettirelim:)) Dans Eden Robot:**

1. Robotumuz için bir fonksiyon oluşturalım.
2. Bu fonksiyonda üç farklı komut olsun.
3. Komutları; İLERİ, GERİ ve SALLAN olarak seçelim.
4. Robotumuz'a enerji verdikten sonra aşağıdaki kartları sırası ile okutalım.

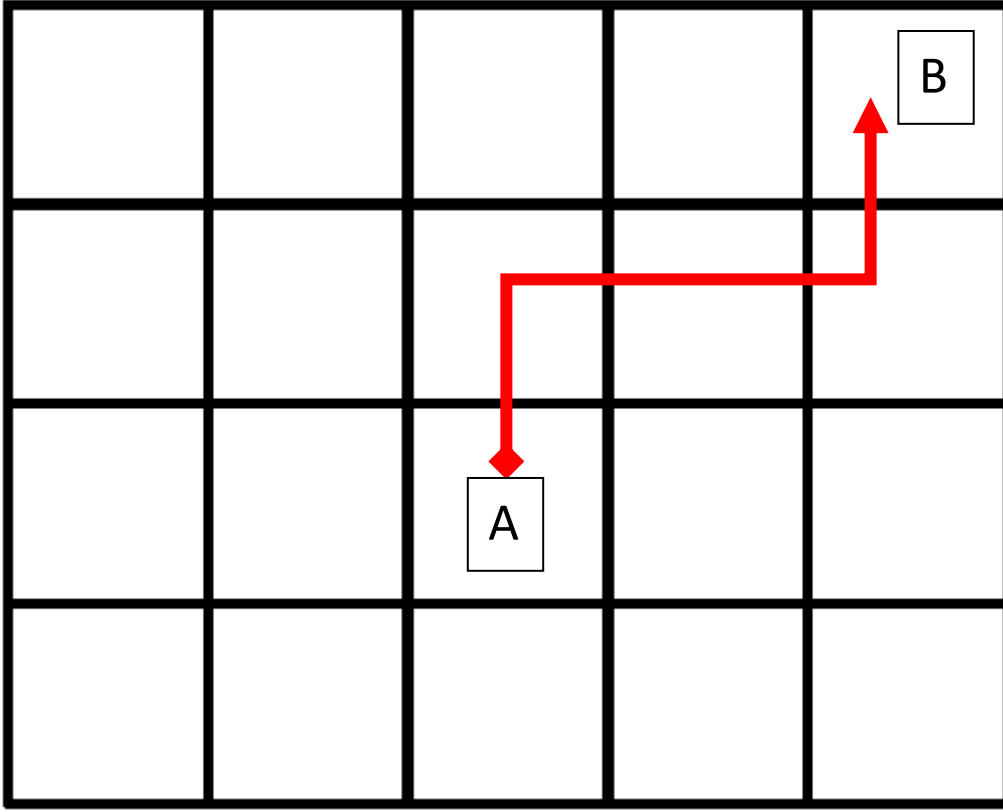


5. Böylece program yazmaya başlamadan fonksiyon tanımlama işlemini gerçekleştirdik.
6. Şimdi aşağıdaki kod dizisini okutarak, programımızı yazalım;
7. En son PLAY kartını okutarak programı çalıştıralım.



- 1. PLAY Kartı'nı okuttuktan sonra program işlemeye başlar.
- 2. Müzik çalar.
- 3. İleri, Geri gider ve Sallanır.
- 4. 360° kendi etrafında döner.
- 5. Program sona erer.

UYGULAMA YAPRAKLARI-1

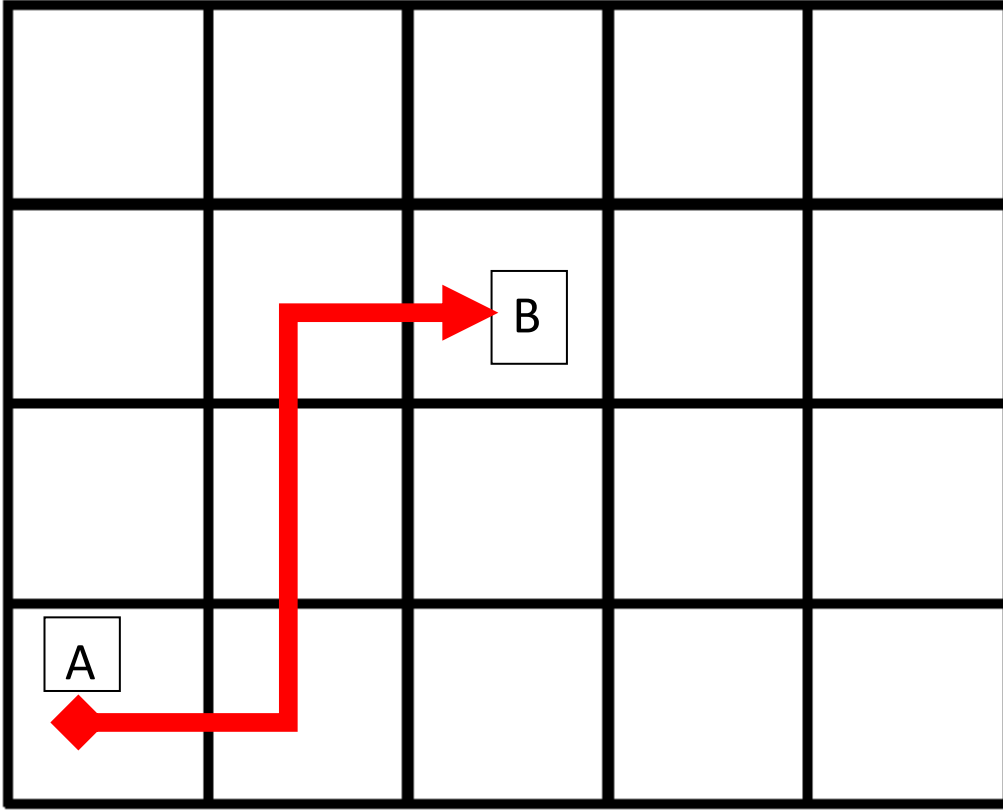


Robotumuzun A karesinden B karesine gitmesi için gerekli olan Kod Dizisini aşağıdaki kutulara sıralayınız .

CEVAP

↑	↶	↑	↑	↷	↑
---	---	---	---	---	---

UYGULAMA YAPRAKLARI-2



Robotumuzun bulunduğu kareden B karesine gitmesi için gerekli olan Kod Dizisini aşağıdaki kutulara sıralayınız.

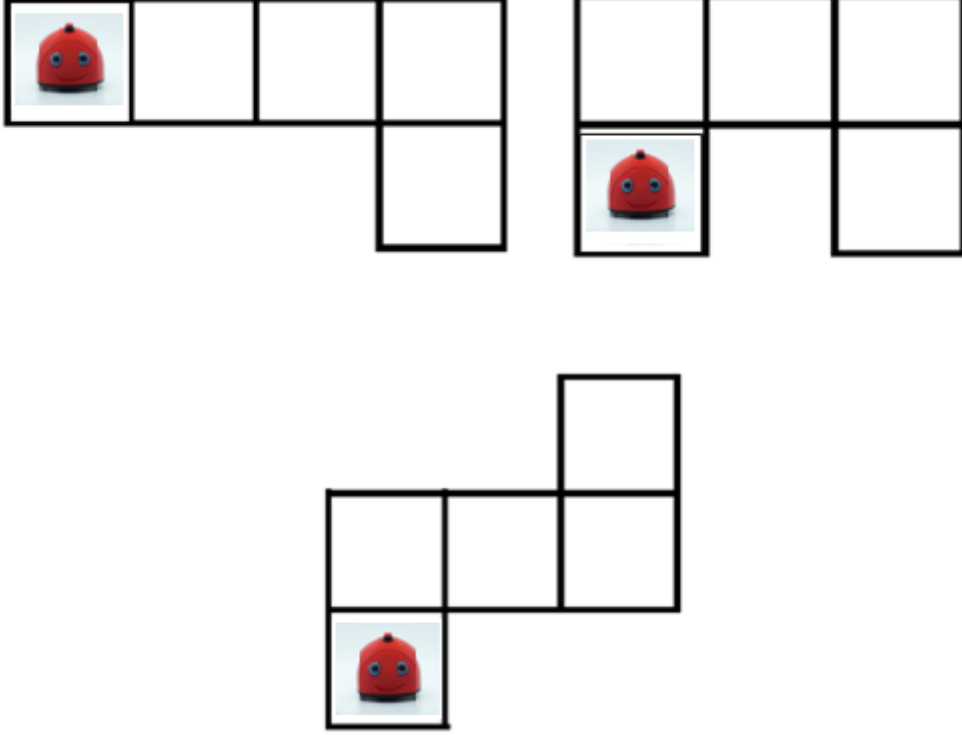
--	--	--	--	--	--	--

CEVAP

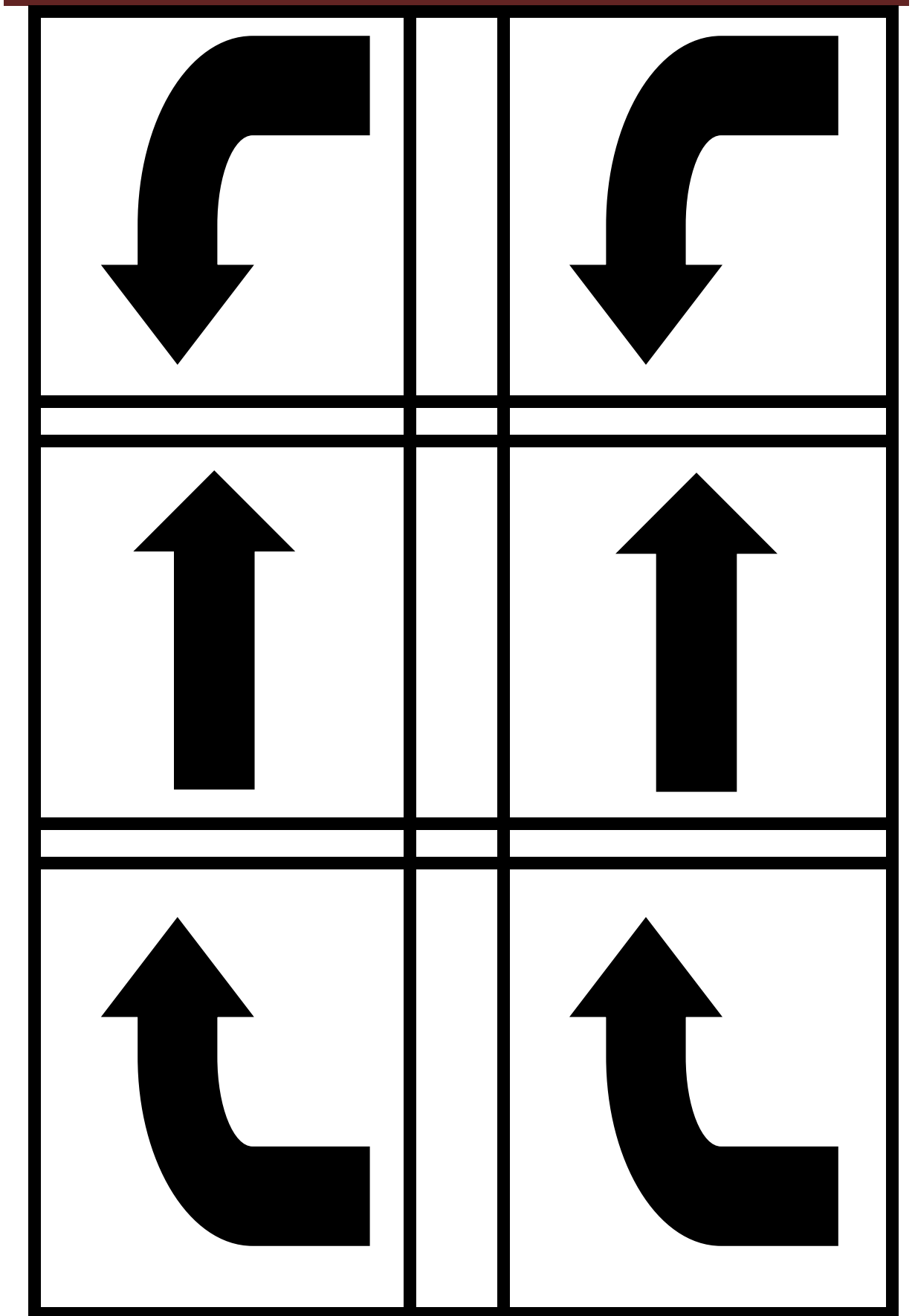
↪	↑	↩	↑	↑	↪	↑
---	---	---	---	---	---	---

UYGULAMA YAPRAKLARI-3

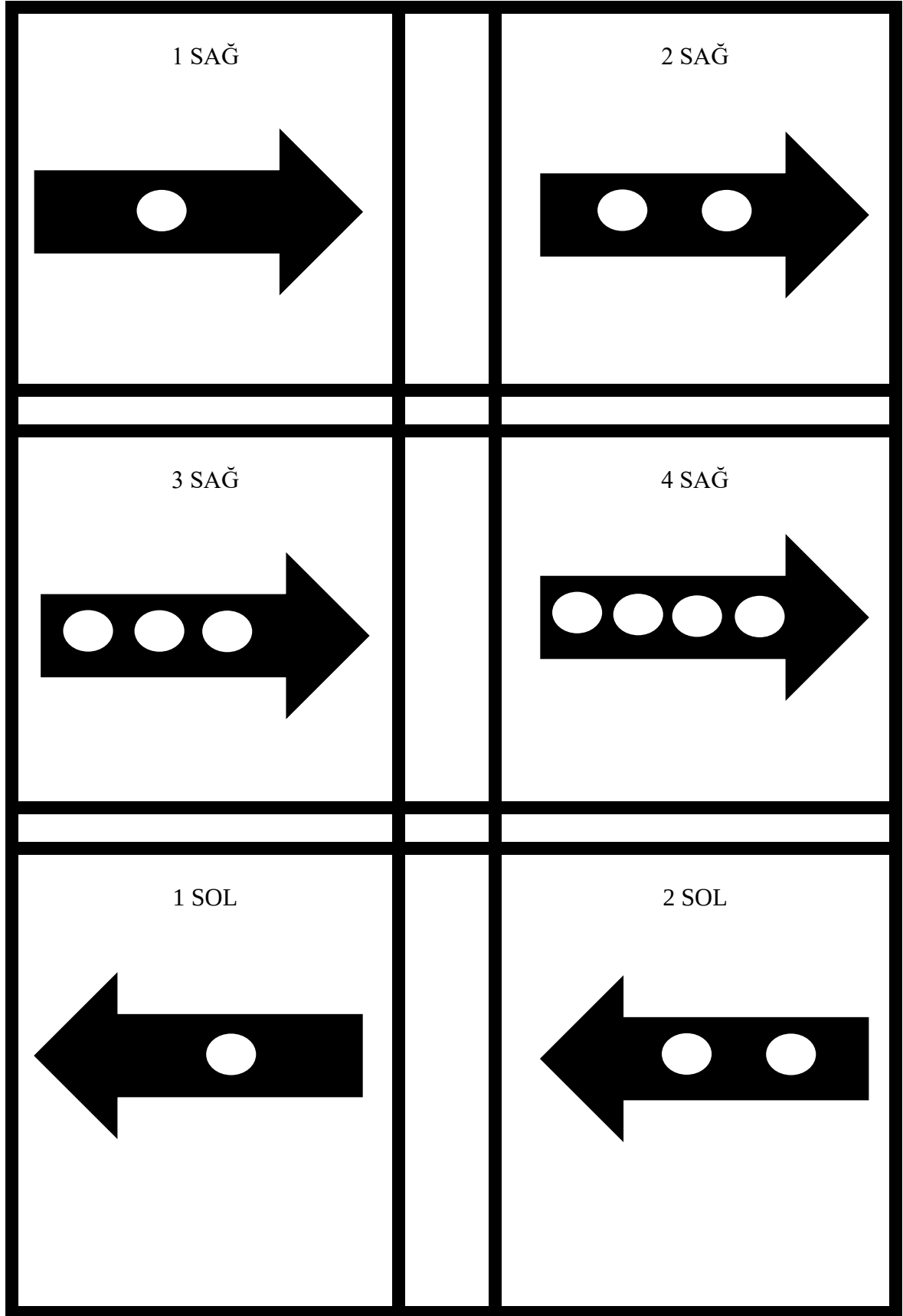
Öğrencilerden, Robotlarını aşağıdaki şekillerde hareket ettirebilmeleri için gerekli olan kodlamaları, önce kağıt üzerinde daha sonra uygulamalı olarak yapmalarını isteyebilirsiniz.



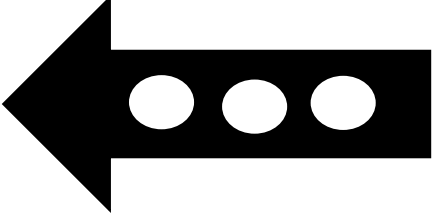
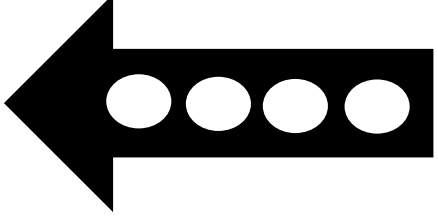
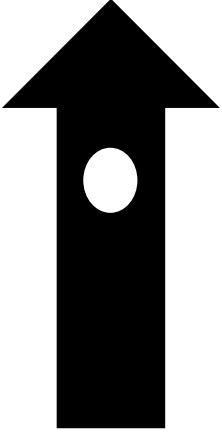
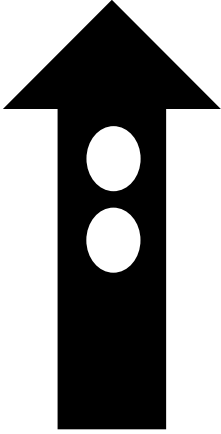
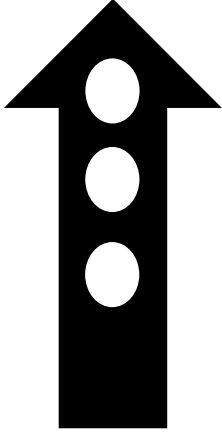
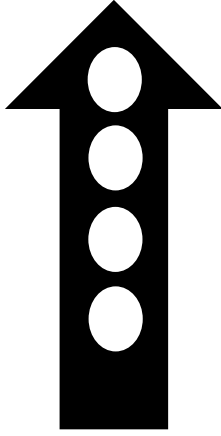
YÖN OKLARI-1



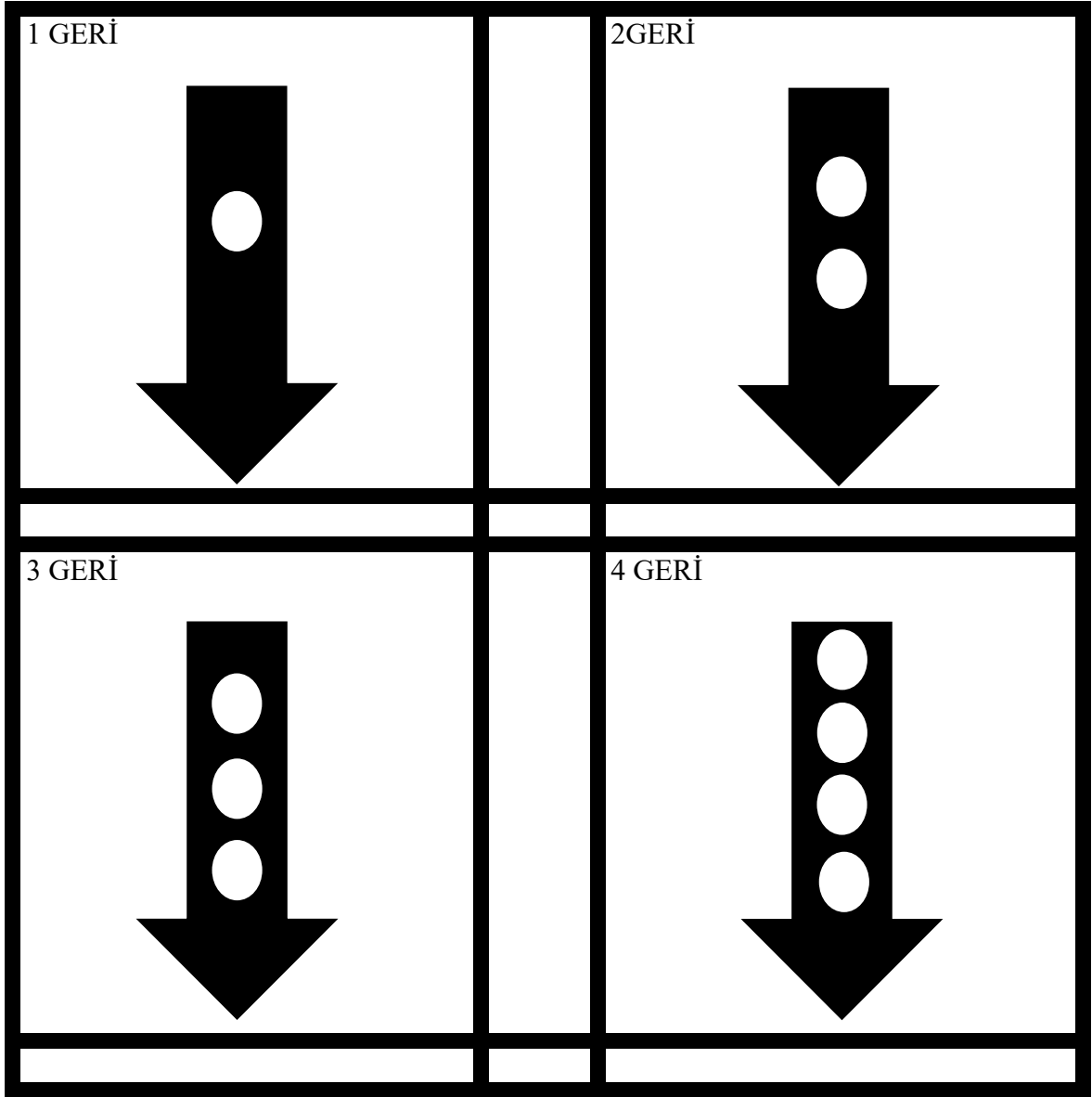
YÖN OKLARI-2



YÖN OKLARI-3

3 SOL 		4 SOL 
1 İLERİ 		2 İLERİ 
3 İLERİ 		4 İLERİ 

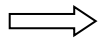
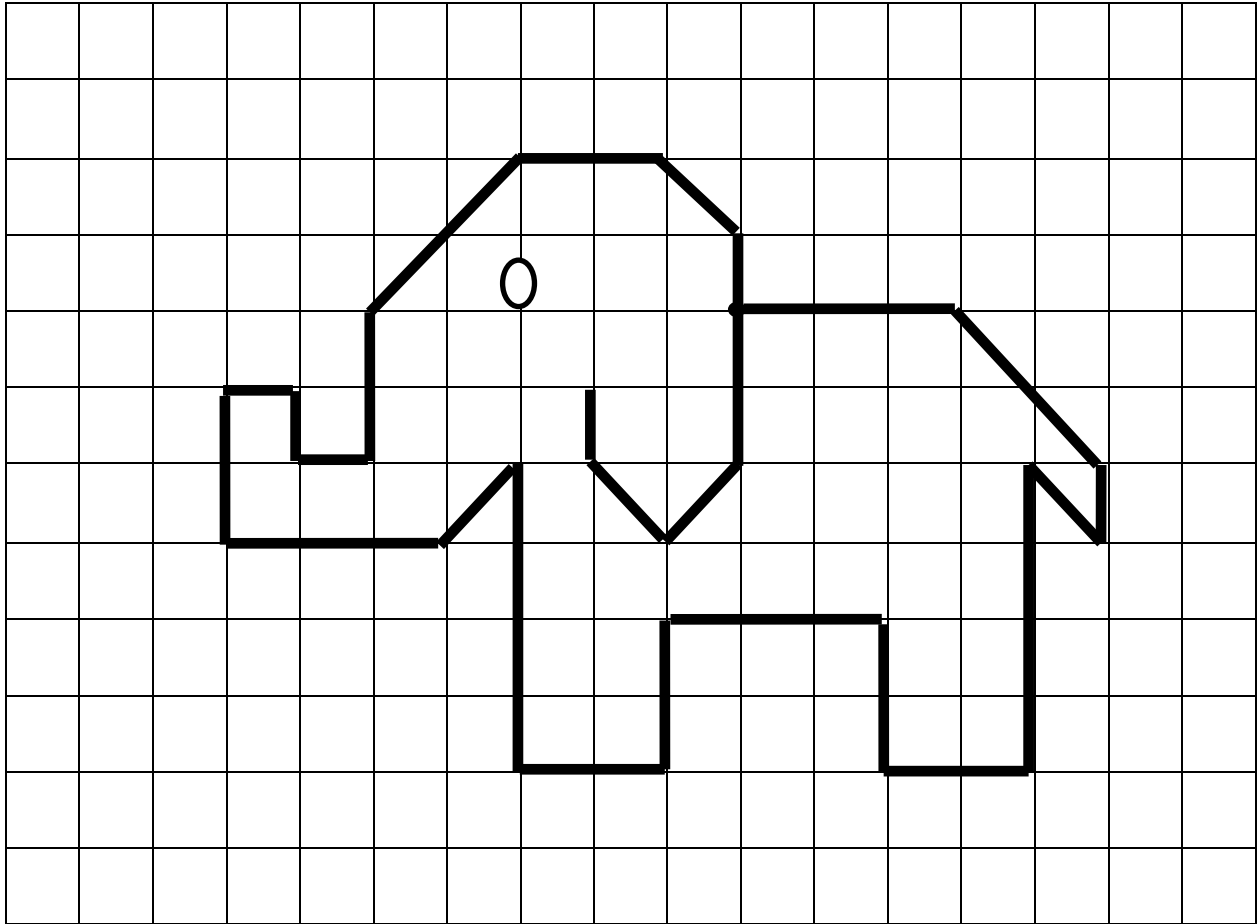
YÖN OKLARI-4



GRAFİK DİKTE ÇALIŞMALARI

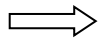
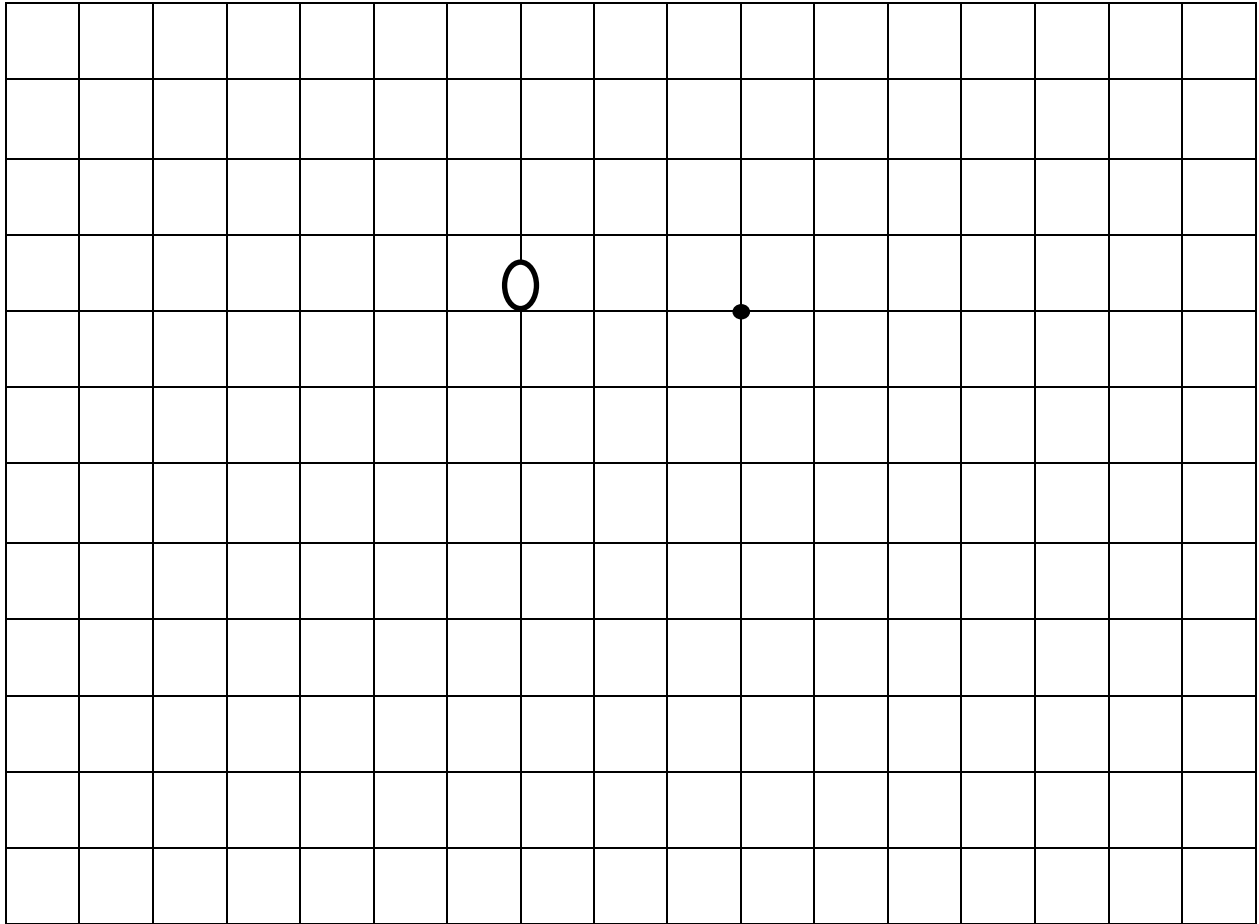
Grafik dikte çalışmaları kodlama eğitiminde kullanılan tekniklerden biridir. Kodlamanın en önemli alt başlıklarından biri Sıralama'dır. Grafik Dikte örneklerinde de doğru sıralama takip edilerek yapılan çizimler sonucunda çeşitli şekiller elde edilmektedir. Örneklerdeki siyah noktadan başlanarak, tüm komutlardaki sayı ve yönler kadar ilerledikten sonra, şekiller meydana çıkacaktır.

FİL (Örnek-1)



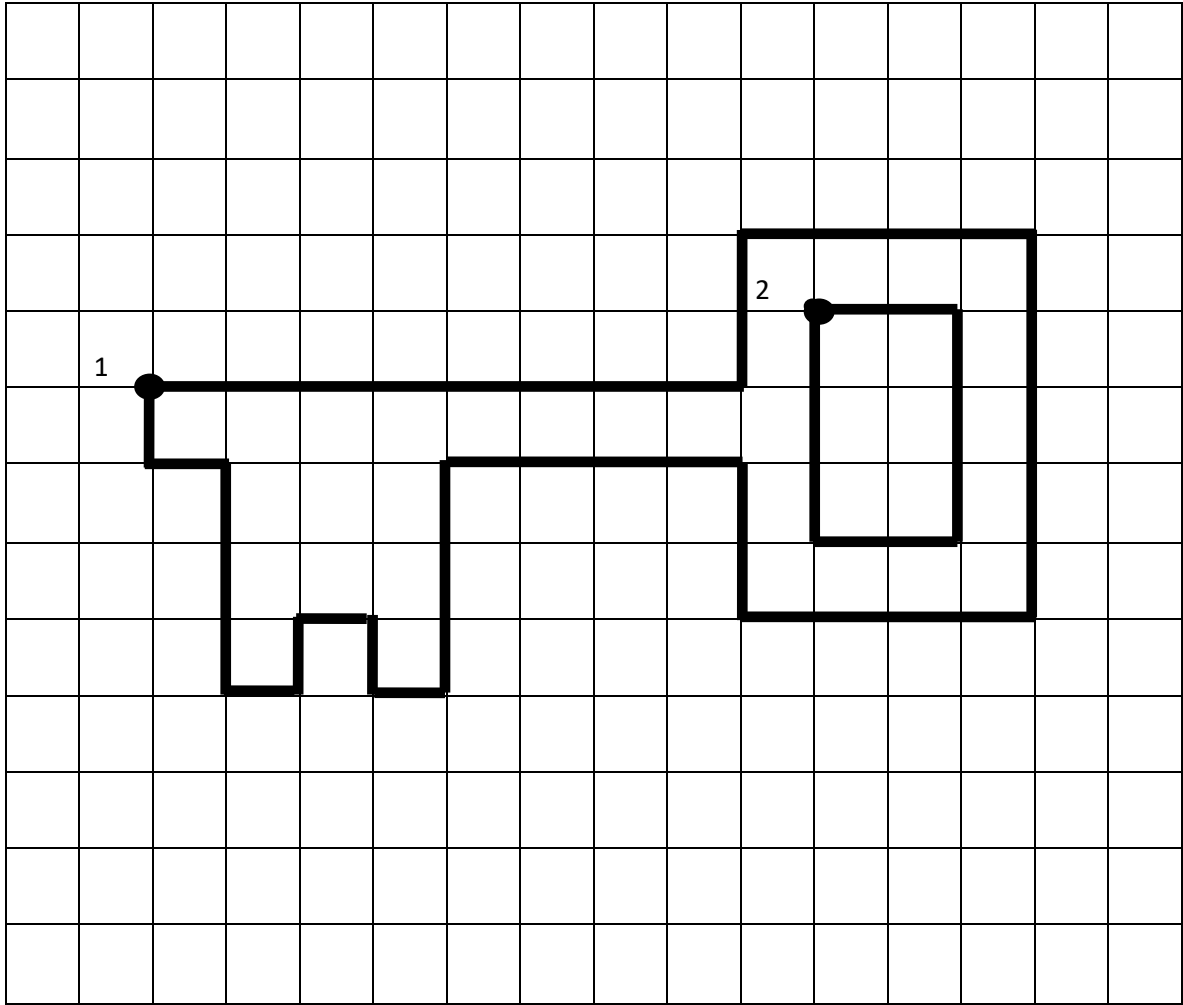
3 →	2 ↘	1 ↓	1 ↖	4 ↓	2 ←
2 ↑	3 ←	2 ↓	2 ←	4 ↑	1 ↗
3 ←	2 ↑	1 →	1 ↓	1 →	2 ↑
2 ↗	2 →	1 ↘	3 ↓	1 ↙	1 ↖
1 ↑					

Örnek-1



3 →	2 ↘	1 ↓	1 ↖	4 ↓	2 ←
2 ↑	3 ←	2 ↓	2 ←	4 ↑	1 ↗
3 ←	2 ↑	1 →	1 ↓	1 →	2 ↑
2 ↗	2 →	1 ↘	3 ↓	1 ↙	1 ↖
1 ↑					

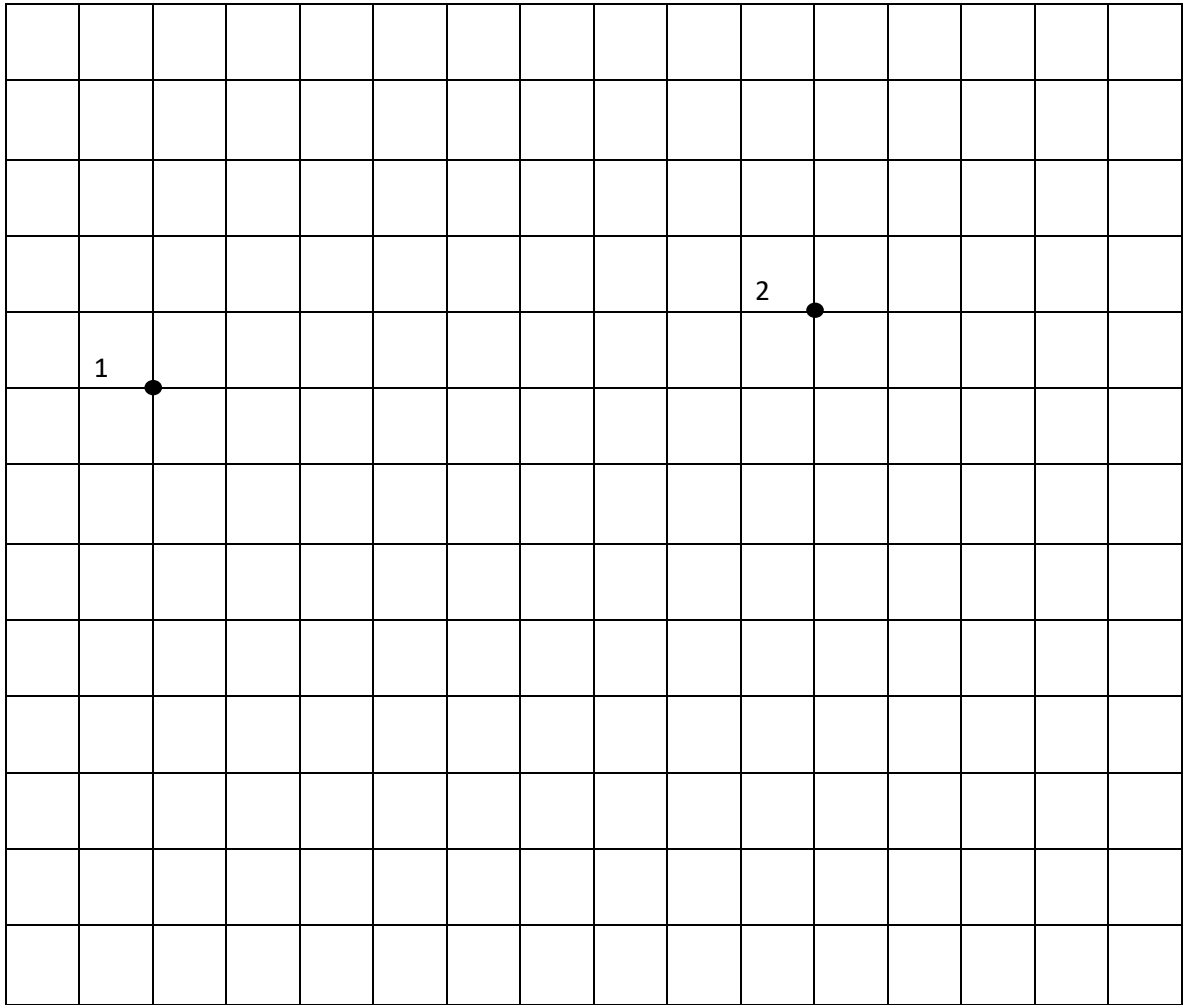
ANAHTAR (Örnek-2)



1	8 →	2 ↑	4 →	5 ↓	4 ←	2 ↑
	4 ←	3 ↓	1 ←	1 ↑	1 ←	1 ↓
	1 ←	3 ↑	1 ←	1 ↑		

2	2 →	3 ↓	2 ←	3 ↑		
---	-----	-----	-----	-----	--	--

Örnek-2



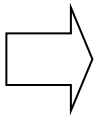
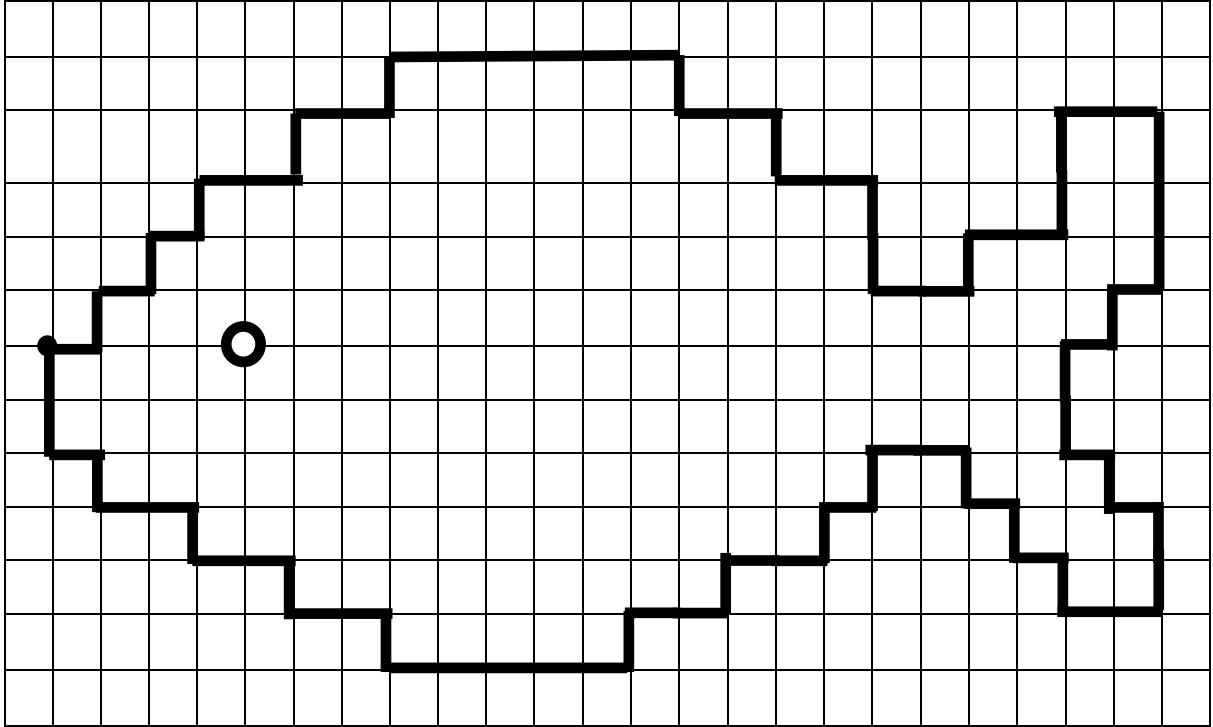
1

8 →	2 ↑	4 →	5 ↓	4 ←	2 ↑
4 ←	3 ↓	1 ←	1 ↑	1 ←	1 ↓
1 ←	3 ↑	1 ←	1 ↑		

2

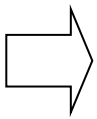
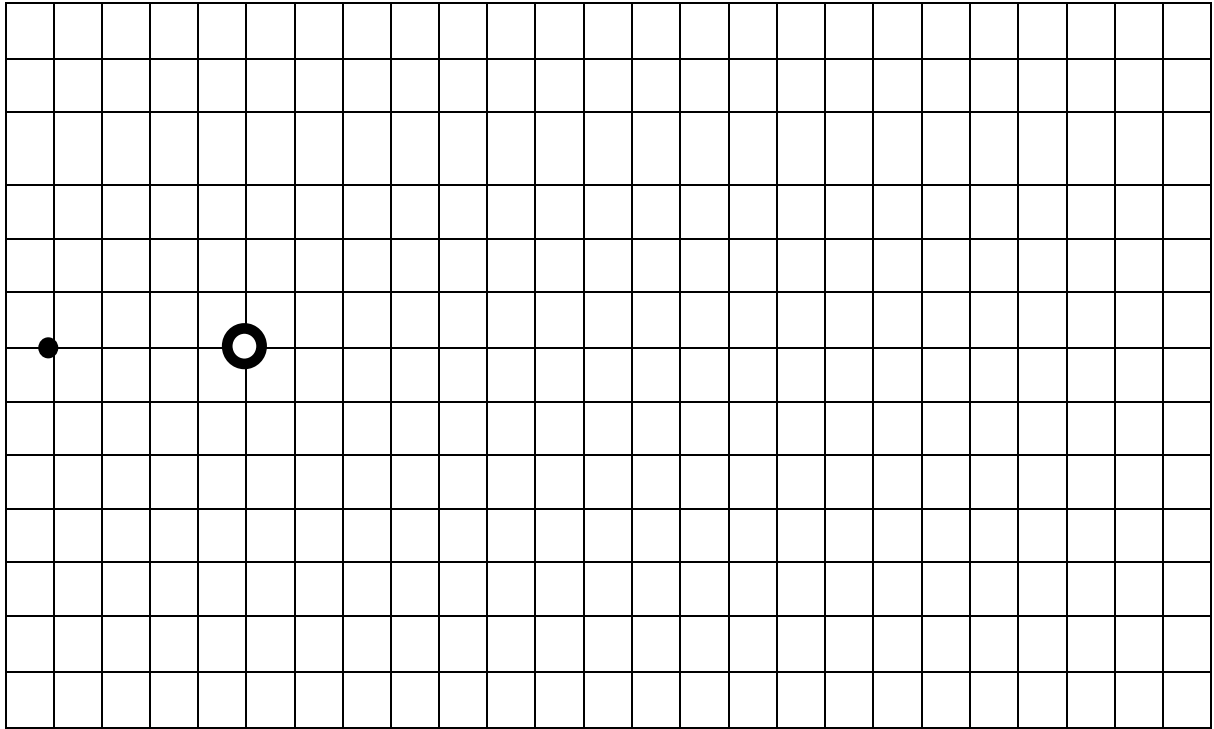
2 →	3 ↓	2 ←	3 ↑		
-----	-----	-----	-----	--	--

BALIK (Örnek-3)



1 →	1 ↑	1 →	1 ↑	1 →	1 ↑
2 →	1 ↑	2 →	2 ↑	6 →	1 ↓
2 →	1 ↓	2 →	2 ↓	2 →	1 ↑
2 →	2 ↑	2 →	3 ↓	1 ←	1 ↓
1 ←	2 ↓	1 →	1 ↓	1 →	2 ↓
2 ←	1 ↑	1 ←	1 ↑	1 ←	1 ↑
2 ←	1 ↓	1 ←	1 ↓	2 ←	1 ↓
2 ←	1 ↓	5 ←	1 ↑	2 ←	1 ↑
2 ←	1 ↑	2 ←	1 ↑	1 ←	2 ↑

Örnek-3



1 →	1 ↑	1 →	1 ↑	1 →	1 ↑
2 →	1 ↑	2 →	2 ↑	6 →	1 ↓
2 →	1 ↓	2 →	2 ↓	2 →	1 ↑
2 →	2 ↑	2 →	3 ↓	1 ←	1 ↓
1 ←	2 ↓	1 →	1 ↓	1 →	2 ↓
2 ←	1 ↑	1 ←	1 ↑	1 ←	1 ↑
2 ←	1 ↓	1 ←	1 ↓	2 ←	1 ↓
2 ←	1 ↓	5 ←	1 ↑	2 ←	1 ↑
2 ←	1 ↑	2 ←	1 ↑	1 ←	2 ↑

