



Numara	Konsept	Faaliyet Adı
D-1	Koşullular (if then)	Sihirli Melodika
D-2	Koşullular (if else)	Ürkek VinciBot
D-3	Koşullular (if else)	Sıkı Dostlar
D-4	Koşullular (if else)	Light Chaser 1.0
D-5	Değişkenler	Spiral Grafikler
D-6	Değişkenler	Maraton
D-7	Değişkenler	Şarj İstasyonu
D-8	Değişkenler	Ödül - Ceza Makinesi
D-9	Değişkenler	Hoş Müzik
D-10	Değişkenler	Catch 3!
D-11	Değişkenler	Sayaç
D-12	Değişkenler	Kronometre
D-13	Fonksiyon (Çoklu Fonksiyon)	Renge bağlı hız değişimi
D-14	Fonksiyon (Çoklu Fonksiyon)	Düzgün Çokgenler
D-15	Fonksiyon (Çoklu Fonksiyon)	Dans eden VinciBot

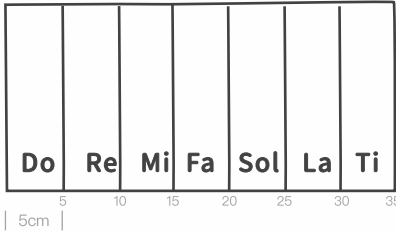
D1 Sihirli Melodika

Koşullular
(if then)



Görev: "AND, OR ve NOT" kodlama bloklarının kullanımının anlamını öğrenin. Sonra bir "Melodika" yapmak için yeni bloklar, ToF menzül sensörü ve müzik blokları kullanın.

- 1 Beyaz kağıda yedi eşit mesafe çizin (Önerilen mesafe 5 cm'dir; Ancak mesafe, mevcut oyun alışkanlıklarına göre ayarlanabilir). Her boşluğa Do, Re, Mi, Fa, Sol, La ve Ti (Si) yazın.

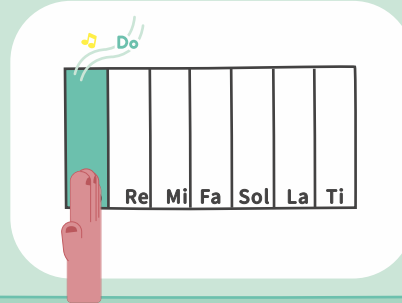
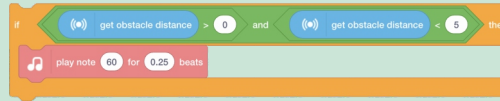


- 2 "AND, OR ve NOT" kodlama bloklarının kullanımının anlamını öğrenin.



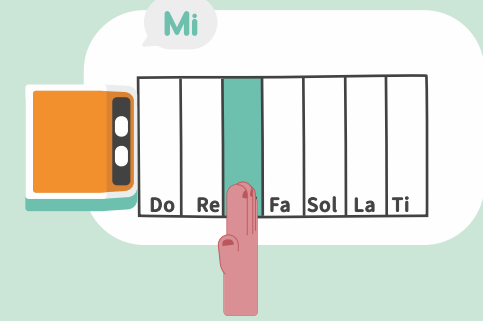
Sırasıyla "AND, OR ve NOT" işlemleri, her iki taraftaki koşulların aynı anda sağlandığını gösterir ("AND"); koşullardan biri karşılandığında ("OR") bu koşullar karşılanmadığında ("NOT") komutunu uygulayın.

- 3 ToF mesafe sensörü farklı mesafe aralıklarını ölçtüğünde, farklı notalar çalınır.

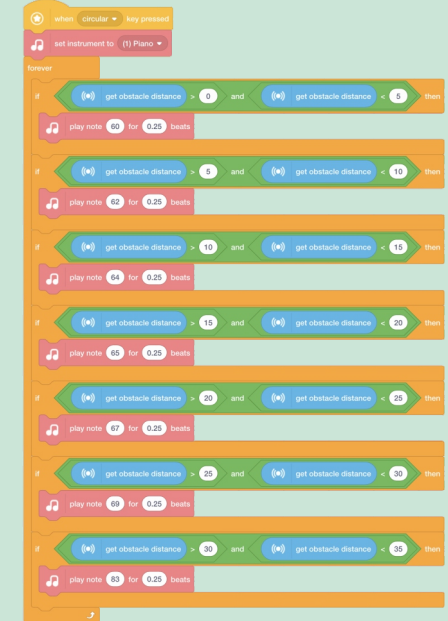


Bonus: Melodika çalarken, ilgili nota Matris ekranında görüntülenmelidir.

- 4 Ellerinizi "Tuşlar"a koyun ve piyanoda güzel bir şarkı çalın.



- 5 Demo Program



D2 Ürkek VinciBot

Koşullular
(if else)



Görev: Koşullu ifadelerde ve program akış şemalarında "if else" ifadesini nasıl kullanacağınızı öğrenin. Ürkek VinciBot tüm dikkatiyle ilerliyor. Bilinmeyen bir engelle karşılaştığında "korkudan" geri çekilecek, sağa dönecek ve ardından gözlerini tüm dikkatiyle ilerlemeye devam edecektir.

1 VinciBot, Matris ekranda görüntülenen iri gözlü bir ifadeyle ilerlemeye devam ediyor.

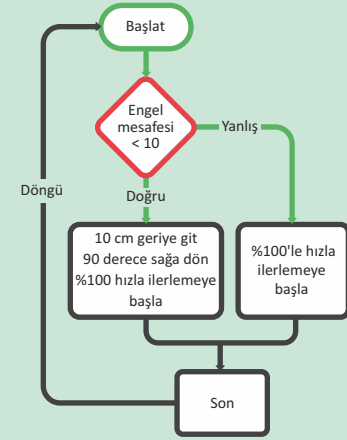
```
when triangle key pressed
start moving forward with 100 % speed
show image [Matrix Eyes]
```



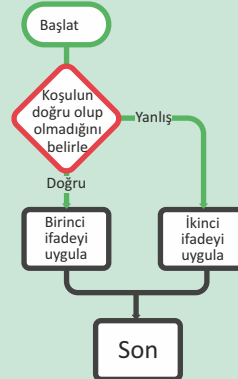
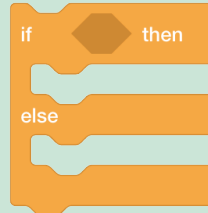
2 İlk olarak, programlamak için "if else" ifadesini kullanın. VinciBot bir engelle karşılaşır "korku" sesi çıkararak ve korkmuş bir ifadeyle 10 cm geriye doğru hareket edecek; sonra sağa dönecek ve gözleri iri iri ilerlemeye devam edecektir.

```
forever
if (get obstacle distance < 12) then
  sound emotion fear
  show image [Matrix Eyes]
  move backward for 10 cm
  turn right for 90 degrees
  start moving forward with 100 % speed
  show image [Matrix Eyes]
```

4 Programı "if else" ifadesini kullanarak yeniden yazın.

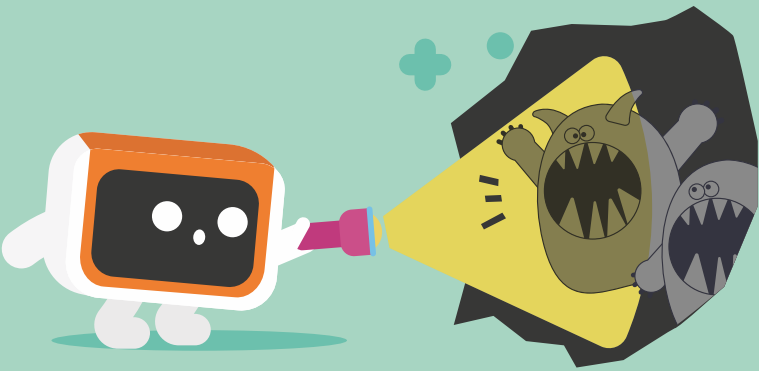


3 "if else" ifadesini kullanmayı öğrenin.



"if else" ifadesinin yapısının uygulayış şeması.

```
when triangle key pressed
forever
if (get obstacle distance < 12) then
  sound emotion fear
  show image [Matrix Eyes]
  move backward for 10 cm
  turn right for 90 degrees
  start moving forward with 100 % speed
  show image [Matrix Eyes]
else
  start moving forward with 100 % speed
  show image [Matrix Eyes]
```



D3 Sıkı Dostlar



Görev: VinciBot'un küçük ayıcı takip etmesini sağlamak için ToF menzil sensörünü kullanın. Küçük ayı aniden kaldırıldığında VinciBot hareket etmeyi bırakır ve "Neredeydin?" diye sorar. Sonra küçük ayı geri geldiğinde VinciBot onu takip etmeye devam ediyor.

Koşullular
(if else)

- 1 Küçük bir ayıcık hazırlayın (veya başka bir oyuncak) ve vinciBot'a yakın bir yere yerleştirin.

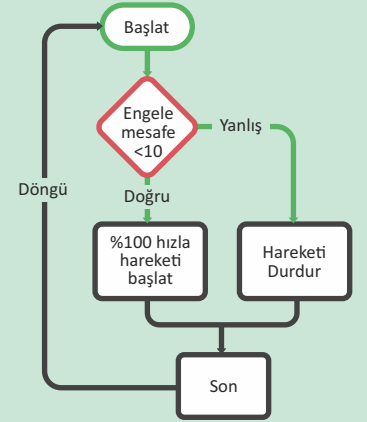


- 2 VinciBot ayıya belirli bir mesafedeyken, yavaşça ayıcığı yaklaştırın. VinciBot duracak ve "Neredeydin? Lütfen beni bekle" diyecek.

```
if ((get obstacle distance < 10) then
  start moving forward with 100 % speed
else
  stop moving
  say "Where have you been? Please wait for me" until done
```

Not: Ayıcığı elinizle yavaşça ilerletin.

- 3 VinciBot her "Lütfen beni bekle" dediğinde, ayıcık tekrar VinciBot'a doğru hareket edecektir. VinciBot da tekrar ayıcığa doğru hareket edecektir. Böylece "forever" kodlama bloğu kullanılmış olacaktır.



```
when triangle key pressed
forever
  if ((get obstacle distance < 10) then
    start moving forward with 100 % speed
  else
    stop moving
    say "Where have you been? Please wait for me" until done
```



Bonus: Yeni bir programlama yaz. VinciBot ayıcığı takip ettiğinde, ayıcıkla VinciBot arasındaki mesafe gerçek zamanlı olarak ekranda gösterilsin.

D4 Light Chaser 1.0

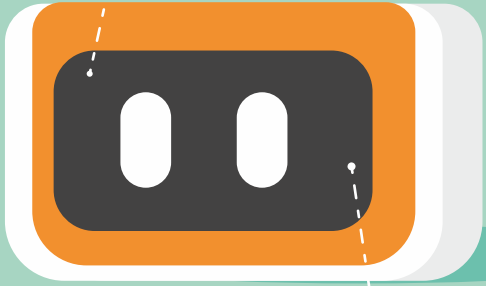
Koşullular
(if else)



Görev: VinciBot'u güçlü bir ışık kaynağını takip edecek şekilde programlamak ve kontrol etmek için ışık algılama sensörünü kullanın. Güçlü ışık kaynağı kaybolduğunda, VinciBot hareket etmeyi durduracaktır.

- 1 Matris ekranın arkasında, VinciBot'un önünde ortam ışığındaki değişiklikleri algılayabilen iki ışık algılama sensörü (hem sol hem de sağ tarafta) vardır. Ortam ışığının değeri 0 ile 100 arasındadır.

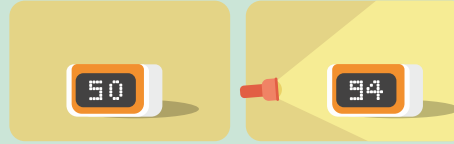
Işık algılama sensörü (Sol)



Light detection sensor (left)

- 2 Normal ortam ışığı altında bir taraftaki ortam ışığının değerini test eden bir program yazın. Ardından el feneri gibi güçlü bir ışık kaynağı hazırlayın. El fenerini açın, doğrudan VinciBot'un önüne doğrultun ve ortam ışığı değerini tekrar test edin. (Not:Bu aktivite mümkün olduğunca 40-60 arası normal ortam ışık değerlerinde yapılmalıdır.)

```
when triangle key pressed
forever
  write ((left ambient light
```



- 3 Normal ve güçlü ışıkta ortam ışığı değerlerine dayalı olarak bir sınır değeri (iki ortam ışığı değeri arasında bir ara değer) belirleyin.

50 60 94

Sınır Değer

- 4 Bir program yazın: Ortam ışığı, arabanın belirli bir değerinden büyük olduğunda, VinciBot güçlü ışığı takip etmeye devam eder; aksi halde durur.

```
if ((left ambient light > 60) then
  start moving forward with 100 % speed
else
  stop moving
```

- 5 Demo Program

```
when triangle key pressed
forever
  if ((left ambient light > 60) then
    start moving forward with 100 % speed
  else
    stop moving
```

D5 Spiral Grafikler

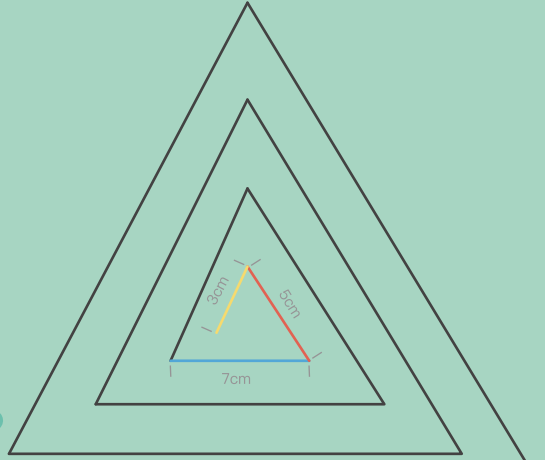
Değişkenler



Görev: VinciBot ile spiral grafiklerin yapısını ve özelliklerini inceleyin; değişkenleri inceleyin ve VinciBot'a programlama yoluyla değişkenleri kullanarak spiral grafiği çizdirin.



1 Spiral grafiğin bir özelliği, kenar uzunluğunun çizim işlemi sırasında sürekli olarak değişmesidir.



2 Değişkeni grafiğin kenar uzunluğunu temsil edecek şekilde ayarlayın.

Değişkenler

Değişken oluştur

Liste yap Bloklarım

My Blocks

Bloklar oluştur

Yeni değişken

Yeni değişken ismi

Kenar uzunluğu

Tüm bağımsız görüntüler için
 Sadece bu bağımsız görüntü için

Çıkış OK

set Side length to 0

change Side length by 1

Değişken bloğunun değeri program çalıştığında değişecektir.

3 Grafiklerin başlangıç kenar uzunluğunu ayarlayın.

when triangle key pressed

set Side length to 3

Bonus: Kare spiral bir grafik çizin. Kare spiral grafik çizmek için neden iç içe döngüler gereklidir?

4 Spiral grafiğin, grafik şeklinin uzama sürelerinin ve kenar uzunluğunun artış değerini ayarlayın.

repeat 10 Uzama zamanı

move forward for Side length cm

turn left for 120 degrees Grafik şekli

change Side length by 2 Yan uzunluğun artış değeri

5 Üçgen ve Beşgen spiral grafik çizin.



Triangle spiral graphic

Pentagonal spiral graphic

when triangle key pressed

set Side length to 3

repeat 10

move forward for Side length cm

turn left for 120 degrees

change Side length by 2

when circular key pressed

set Side length to 2

repeat 20

move forward for Side length cm

turn left for 72 degrees

change Side length by 0.5

D6 Maraton

Değişkenler



Görev: VinciBot bir maraton yarışına katılıyor. Her 5 santimetre ileri hareket ettiğinde, bir piksel (enerji) bloğu kapatılacaktır. Nokta vuruşlu ekran karardığında, VinciBot'a "Çok yorgunum" dedirtin, kırmızı LED ışıklarını açın ve üç saniye sonra söndürün.

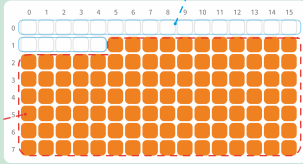
- 1 Yarışmadan önce, VinciBot'u tüm piksel bloklarının yanması ve yeşil LED ışıklarının yanması için programlayın.

```
when triangle key pressed
  show image
  set all LEDs to color
```

- 3 Bir sıradaki piksel (enerji) blokları tükendiğinde, sonraki sıradaki piksel (enerji) blokları da tüketilecektir.

```
if x > 15 then
  change y by 1
  set x to 0
```

X 15ten büyük olduğunda, sıradaki enerji piksel blokları tükenir.

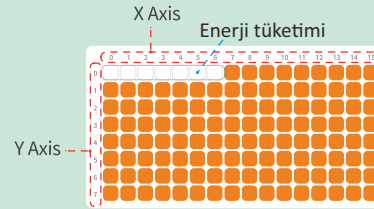


When y+1, x restarts from 0.

- 2 VinciBot her 5 santimetre ileri hareket ettiğinde, bir piksel (enerji) bloğu kapatılacaktır: her piksel bloğunun koordinatlarını temsil eden iki değişken (x,y) ayarlayın.

```
set x to 0
set y to 0
```

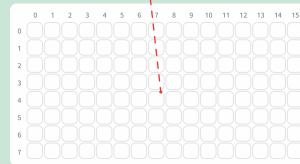
```
move forward for 5 cm
set pixel brightness at x: x y: y as 0 %
change x by 1
```



- 4 Tüm piksel (enerji) blokları tükendiğinde, VinciBot "çok yoruldum" der, kırmızı LED ışıklarını açar ve üç saniye sonra kapatır.

```
if y > 7 then
  say I am so tired
  set all LEDs to color
  wait 3 seconds
  turn all LEDs off
  stop all
```

Tüm piksel enerji blokları tükendi.



- 5 Demo Program

```
when triangle key pressed
  show image
  set all LEDs to color
  set x to 0
  set y to 0
  forever
    move forward for 5 cm
    set pixel brightness at x: x y: y as 0 %
    change x by 1
    if x > 15 then
      change y by 1
      set x to 0
    if y > 7 then
      say I am so tired
      set all LEDs to color
      wait 3 seconds
      turn all LEDs off
  stop all
```



Bonus: Yalnızca iki sıra piksel (enerji) bloğu kaldığında, düşük enerjiyi belirtmek için sarı LED ışıkları açın.



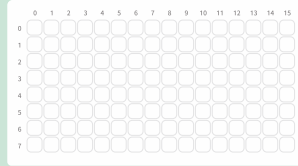
D7 Şarj İstasyonu

Değişkenler

Görev: VinciBot'un enerjisi maraton sırasında tükendi ve acil durumda şarj edilmesi gerekiyor; üçgen düğmesine bastıktan sonra VinciBot "düşük pil, şarj etmeye başlayın" der ve matris ekranındaki piksel (enerji) blokları yavaş yavaş yanar; matris ekran tamamen yandığında şarj işlemi durmalı ve VinciBot "pil tamamen şarj oldu" demelidir.

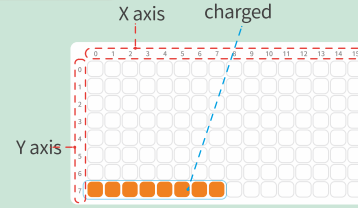
- 1 VinciBot maraton boyunca enerjisi tükendi ve şarj edilmesi gerekiyor.

```
when triangle key pressed
  show image
  set all LEDs to color
  say "The battery is low and needs to be charged" until done
```



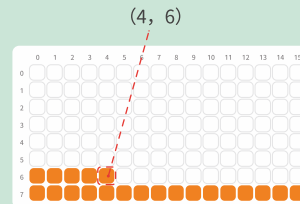
- 2 Her 0,2 saniyede bir, piksel (enerji) bloğu yanacaktır. Her piksel (enerji) bloğunun koordinatlarını temsil edecek iki değişken (x,y) ayarlanmalıdır.

```
set x to 0
set y to 7
forever
  set pixel brightness at x: x y: y as 100 %
  wait 0.2 seconds
  change x by 1
```



- 3 Bir sıra piksel (enerji) bloğu tamamen yandığında ($x > 15$), yeni bir sıradan başlamak gerekir; yani x, 0'a çevrilir ve y, 1 azaltılır. Tüm piksel (enerji) blokları yandığında, yani $y < 0$, şarjı durdurun.

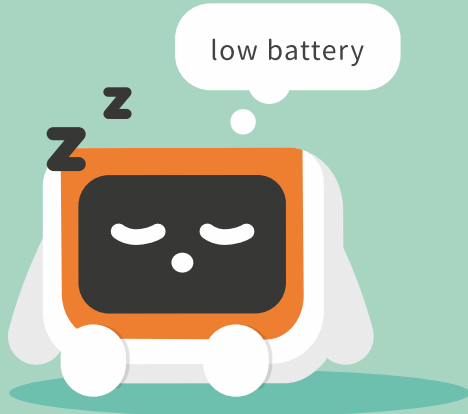
```
if x > 15 then
  set x to 0
  change y by -1
if y < 0 then
  set all LEDs to color
  say "The battery is full" until done
stop all
```



- 4 Demo program

```
when triangle key pressed
  show image
  set all LEDs to color
  say "The battery is low and needs to be charged" until done
set x to 0
set y to 7
forever
  set pixel brightness at x: x y: y as 100 %
  wait 0.2 seconds
  change x by 1
  if x > 15 then
    set x to 0
    change y by -1
  if y < 0 then
    set all LEDs to color
    say "The battery is full" until done
  stop all
```

Bonus: İlk altı sıranın piksel (enerji) blokları yandığında şarj hızı yarı yarıya azalır. Bu arada, tüm piksel (enerji) blokları yanana kadar mavi LED ışıklar yanar. Mavi LED ışıklar daha sonra yeşile dönecektir.



D8 Ödül ve Ceza Makinesi

Değişkenler



Görev: VinciBot'u bir parti aracına, bir ödül ve ceza makinesine dönüştürün! Üç mod ayarlayın: VinciBot için "Pass", "Reward" ve "Punish". Düğmeye basıldığında rastgele bir durum oluşur.

- 1 1 ile 10 arasında rastgele bir değer alacak bir değişken ayarlayın.

```
set X to pick random 1 to 10
```

- 2 Üç modun olasılığını ayarlayın ("Pass", "Reward" ve "Punish"). Üç modun etkilerini tasarlamak için ifadeleri, LED ışıkları, ses efektlerini ve diğer kodlama bloklarını kullanın.

```
if X > 2 and X < 8 then
  show image [game score]
  sound game score
  set all LEDs to color [blue]
```

```
if X < 3 then
  show image [game upgrade]
  sound game upgrade
  set all LEDs to color [green]
```

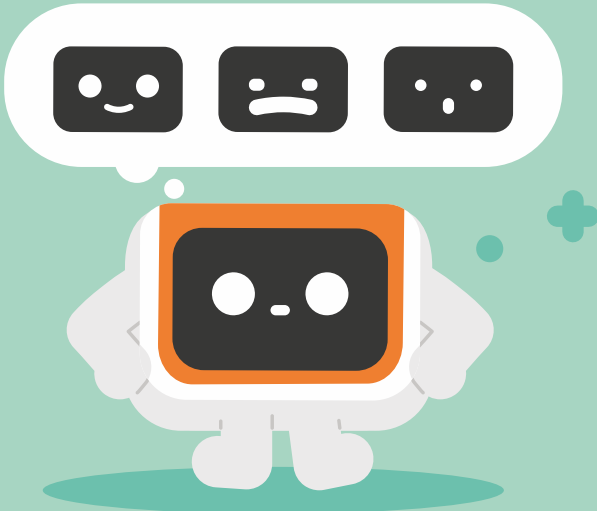
```
if X > 7 then
  show image [game warning]
  sound game warning
  set all LEDs to color [red]
```

- 3 Demo Program

```
when triangle key pressed
  set X to pick random 1 to 10
  if X < 3 then
    show image [game upgrade]
    sound game upgrade
    set all LEDs to color [green]
  if X > 2 and X < 8 then
    show image [game score]
    sound game score
    set all LEDs to color [blue]
  if X > 7 then
    show image [game warning]
    sound game warning
    set all LEDs to color [red]
```

- 4 Oyunu başlatmak için düğmeye basın ve kimin daha şanslı olduğunu karşılaştırın! Örneğin: "Pass" güvenlik anlamına gelir, "Reward" bir gösteriyi gerçekleştirmesi için bir kişiyi atayabileceğiniz anlamına gelir ve "Punish" Doğruluk mu Cesareti mi tamamlamanız gerektiği anlamına gelir.

- 5 Üç modun olasılığını değiştirin ve istatistiksel oyunun sonuçlarını birkaç oyun üzerinden karşılaştırın!



D9 Hoş Müzik

Değişkenler



Görev: VinciBot müzik dinliyor. VinciBot'u, nokta vuruşlu ekrandaki piksel bloklarının müzikle birlikte yükselip alçalacağı şekilde programlayın.

- 1 Üç farklı yükseklikte ses şeması, farklı ses seviyelerine karşılık gelen piksel bloklarının görüntülenmesiyle tasarlanmıştır.



- 2 Algılanan ses seviyesini belirtmek için "Volume Level" değişkenini ayarlayın.

```
set Volume Level to loudness
```

- 3 Önce çalınacak müziğin yaklaşık ses yüksekliği aralığını belirleyin.

```
when triangle key pressed
forever
write loudness
```

01-21



Bonus: Yukarıdaki programa göre, programlamak için "LED ışık" kodlama bloklarını kullanın, böylece LED ışıklar farklı ses hacimlerine göre renk değiştirir.

- 4 Üç ses aralığı ayarlayın ve VinciBot'un farklı ses düzeylerine karşılık gelen farklı ses şemaları göstermesini sağlayın.

```
if Volume size > 0 and Volume size < 8 then
show image [Sound Wave 1]

if Volume size > 7 and Volume size < 15 then
show image [Sound Wave 2]

if Volume size > 14 and Volume size < 21 then
show image [Sound Wave 3]
```

- 5 Demo program

```
when triangle key pressed
turn off screen
forever
set Volume level to loudness
if Volume level > 0 and Volume level < 8 then
show image [Sound Wave 1]
if Volume level > 7 and Volume level < 15 then
show image [Sound Wave 2]
if Volume level > 14 and Volume level < 21 then
show image [Sound Wave 3]
```

D10 Catch 3!

Değişkenler



Görev: 1-20 sayıları VinciBot'un matris ekranında belirli bir süre içinde rastgele görünür. Lütfen rastgele sayıları gözlemleyin ve düğmeye basarak 3 sayısını veya 3'ün katlarını "yakalayın". Son olarak, kaç tane 3 yakaladığınızı gözlemleyin.



- 1 Oyun başlıyor! VinciBot, 1'den 20'ye kadar olan sayıları rastgele görüntüler ve sayılar arasındaki aralık süresini ayarlar.

```
when triangle key pressed
forever
  set Number to pick random 1 to 20
  write Number
  wait 0.5 seconds
```



- 2 Puanlama mekanizmasını ayarlayın: görüntülenen sayı 3 veya 3'ün katı olduğunda, hızlıca kare düğmeye basın. 3 zamanında başarılı bir şekilde yakalandığında, bir puan kazanılır. Aksi takdirde, bir puan düşülür.

```
if (key pressed) is square key pressed? then
  if Number mod 3 = 0 then
    change Score by 1
    sound game score until done
  else
    change Score by -1
    sound game wrong until done
```

+1 point



- 3 Oyun süresini ayarlayın; Oyunun sonunda nihai skoru görüntüleyin ve okuyun.

```
when triangle key pressed
  set Score to 0
  set Time to 60
  repeat until Time = 0
    wait 1 seconds
    change Time by -1
  stop other scripts in sprite
  turn off screen
  wait 1 seconds
  set all LEDs to color red
  say Your score is until done
  say Score
  write Score
```

İpucu: Oyun süresi serbestçe ayarlanabilir.



- 4 Demo program

```
when triangle key pressed
  set Score to 0
  set Time to 60
  repeat until Time = 0
    set Number to pick random 1 to 20
    write Number
    wait 0.5 seconds
    if (key pressed) is square key pressed? then
      if Number mod 3 = 0 then
        change Score by 1
        sound game score until done
      else
        change Score by -1
        sound game wrong until done
  stop other scripts in sprite
  turn off screen
  wait 1 seconds
  set all LEDs to color red
  say Your score is until done
  say Score
  write Score
```



Bonus 1: Diğer numaraları tutmayı deneyin.
Bonus 2: Puanlama yöntemini düğmeye basmaktan alkışlamaya değiştirin.

D12 Kronometre

Değişkenler



Görev: VinciBot bir kronometre sayacına dönüşür: A koşulu tetiklendiğinde, zamanlama başlar ve B koşulu tetiklendiğinde süre biter. Koşul C'yi tetiklerken, süre 0'a döner.



- 1 Bir kronometrenin işlevini düşünün ve zamanlama periyodunu temsil etmesi için yeni bir "x" değişkeni oluşturun.

```
when triangle key pressed
set x to 0
write x
```

- 2 Zamanlama başlangıcını (üçgen düğmesine basıldığında) ve zamanlama ekranı için aralık süresini tetiklemek için koşul A'yı ayarlayın.

```
if is triangle key pressed then
wait 1 seconds
change x by 1
write x
```

- 4 Son olarak, zamanlayıcının 0'a sıfırlanması için tetikleme koşulu C'yi (yuvarlak düğmeye basıldığında) ayarlayın.

```
if is circular key pressed? then
set x to 0
write x
```

- 5 Demo program

```
when triangle key pressed
set x to 0
write x
forever
if is triangle key pressed? then
repeat until is square key pressed?
wait 1 seconds
change x by 1
write x
change x by 0
write x
if is circular key pressed? then
set x to 0
write x
```

- 3 Programa "repeat until" ifadesini uygulayın ve başka bir B koşulu tetiklendiğinde (düğmeye basıldığında) duracak şekilde zamanlamayı ayarlayın.

```
if is triangle key pressed? then
repeat until is square key pressed?
wait 1 seconds
change x by 1
write x
change x by 0
write x
```



D11 Sayaç

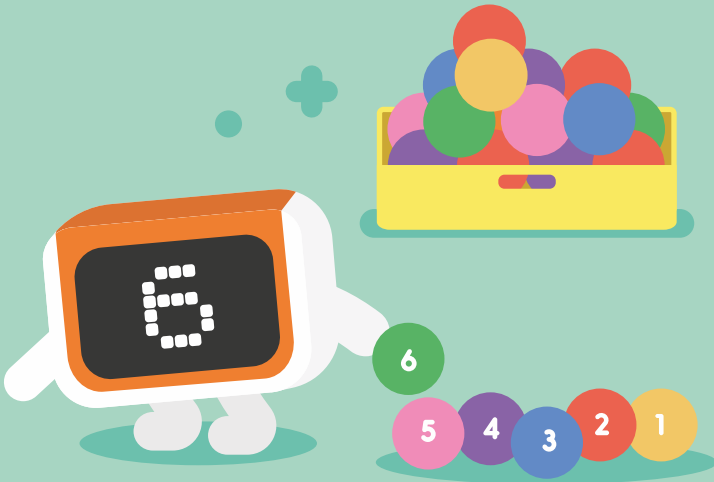
Değişkenler



Görev: VinciBot bir sayaca dönüşür: sayıyı artırmak, azaltmak veya sıfırlamak için farklı düğmelere basın.

- 1 Sayım sayısı olarak yeni bir "sayı" değişkeni oluşturun.

```
when robot start
set Number to 0
write Number
```



- 2 Kalite değişikliğini harekete geçirmek için koşulları ayarlayın: Numarayı artırmak, azaltmak ve sıfırlamak için 3 farklı tuşa basın ve matris ekranında numarayı görüntüleyin. Ayrıca, her tuşa basıldığında LED ışık efektleri ekleyin.

```
if (circular key pressed?) then
set Number to 0
set LEDs array
wait 0.2 seconds
turn all LEDs off
```

```
if (square key pressed?) then
change Number by -1
set LEDs array
wait 0.2 seconds
turn all LEDs off
```

```
if (triangle key pressed?) then
change Number by 1
set LEDs array
wait 0.2 seconds
turn all LEDs off
```

- 3 Sayım fonksiyonunu yerine getirin.

```
when robot start
set Number to 0
forever
write Number
if (triangle key pressed?) then
change Number by 1
set LEDs array
wait 0.2 seconds
turn all LEDs off
if (square key pressed?) then
change Number by -1
set LEDs array
wait 0.2 seconds
turn all LEDs off
if (circular key pressed?) then
set Number to 0
set LEDs array
wait 0.2 seconds
turn all LEDs off
```



Bonus: Farklı sayım senaryolarını göz önünde bulundurun ve renk sayımı, ışık sayımı, ses sayımı vb. gibi yeni sayım harekete geçme koşullarını ayarlayın.

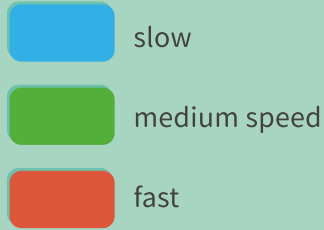
D13 Renge Bağlı Hız Değişimi

Fonksiyon
(Çoklu Fonksiyon)



Görev: Yeni bir "speed" bloğu oluşturun ve yeni bloğa bir girdi olarak "x" parametresini eklemenin bir yolunu bulun. "x" parametresi hızdaki değişimi temsil eder. VinciBot'u farklı renkler algıladığında hızı değiştirecek şekilde programlayın.

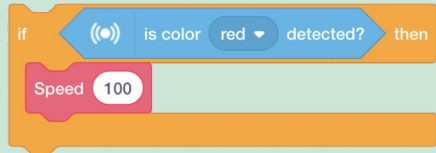
1 Mavi, kırmızı ve yeşil kartlar arka arkaya aralıklarla yerleştirilmiştir ve bu renkli kartlar yoldaki değişen bölgeleri temsil etmektedir.



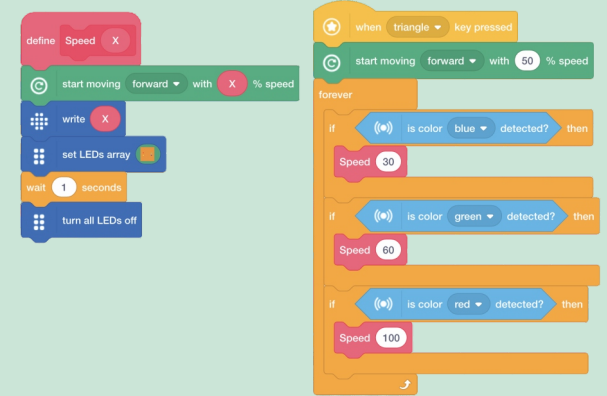
2 Yeni bir blok "speed" yapın ve "speed" parametrelerini ayarlayın. Hız değiştiğinde, LED ışıkları yanıp söner ve mevcut hızı ekranda gösterir.



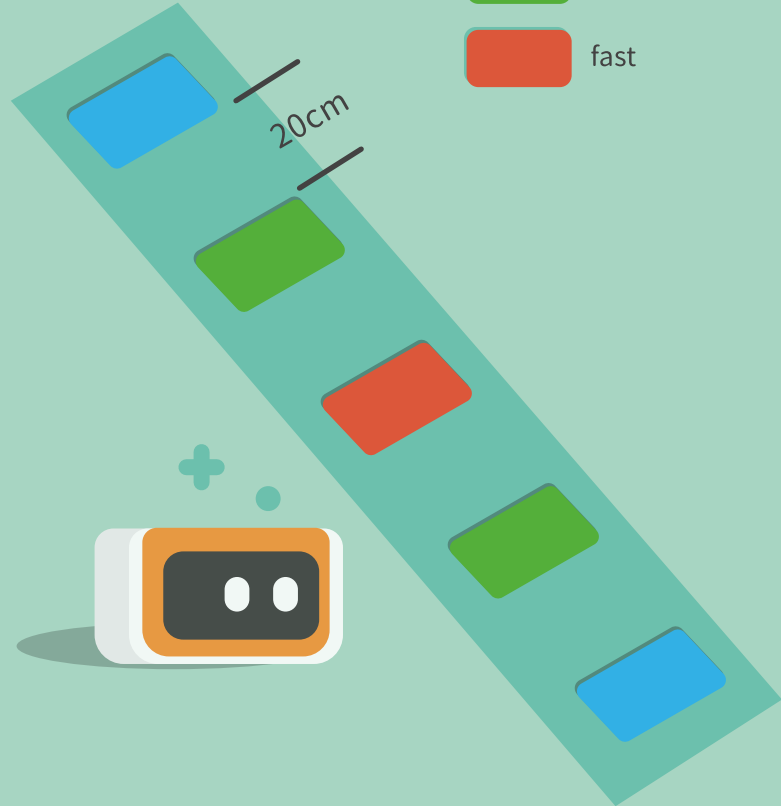
3 Farklı renklendirilmiş bölgeye doğru geçtiğinde VinciBot'a hız uyarı yapın.



4 VinciBot'un geçiş bölgeleri ile yoldan geçmesini sağlayan bir program yazınız.



Bonus: Farklı değiştirme bölgelerinden geçerken ses efektlerini, LED ışıklarını ve diğer efektleri VinciBot'a ekleyin.

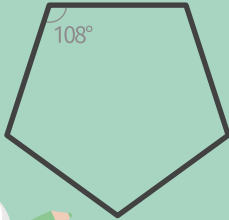
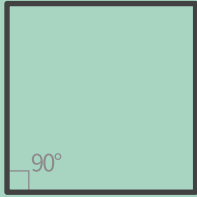
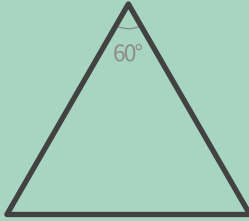


D14 Düzgün Çokgenler



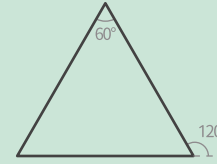
Görev: Yeni bloğa iki parametre ekleyin; iki parametre "number of sides (kenar sayısını)" ve "angle (açıyı)" temsil eder. Bu iki parametreyi değiştirerek, VinciBot çeşitli düzgün çokgenler çizebilir.

1 Düzgün bir çokgende her bir kenarın uzunluğu ve her bir iç açısının ölçüsü aynıdır.



2 VinciBot düzgün bir çokgen çizdiğinde, "number of sides (kenar sayısını)" tekrar sayısına eşittir, "angle (açı)" ise 180 derece eksi iç açının açısına eşittir. Yeni bir blok yapın ve "number of sides (kenar sayısını)" ve "angle (açıyı)" temsil eden iki parametre ekleyin.

```
define Draw a graphic Number of sides Turn Angle
repeat Number of sides
  move forward for 10 cm
  turn left for Angle degrees
```



3 VinciBot'un daha düzenli çokgenler çizmesine yardımcı olmak için farklı parametreleri modifiye edin.

```
Regular pentagon
when circular key pressed
  Draw a graphic 5 Turn 72

define Draw a graphic Number of sides Turn Angle
repeat Number of sides
  move forward for 10 cm
  turn left for Angle degrees

Regular triangle
when triangle key pressed
  Draw a graphic 3 Turn 120

Square
when square key pressed
  Draw a graphic 4 Turn 90
```



Bonus: Bu yöntem düzgün çokgen olmayan bir şekil çizmek için kullanılabilir mi? Neden ya da neden olmasın?



D15 Dans eden VinciBot

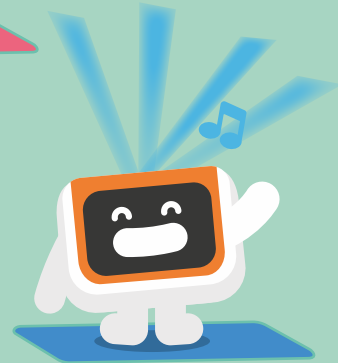
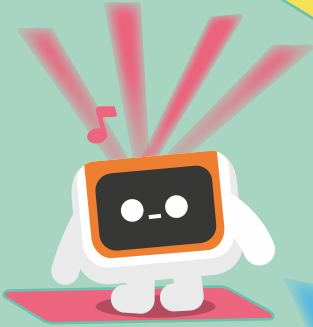
Fonksiyon
(Çoklu Fonksiyon)



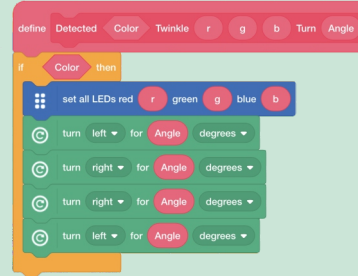
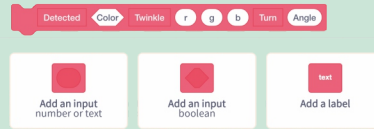
Bonus: Yeni bloğa RGB parametreleri ve değerlendirme koşulları ekleyin ve VinciBot'u, algılandıklarında farklı renklere karşılık gelen LED ışıkları görüntülerken farklı genliklerde salınım hareketleri yapacak şekilde programlayın.



1 VinciBot'u, algılandıklarında farklı renklere karşılık gelen LED ışıkları görüntülerken farklı danslar (farklı genliklerde sallanma hareketleri) gerçekleştirmesi için programlayın.



2 Yeni blokları analiz edin ve VinciBot'un farklı danslar (farklı genliklerde salınım hareketleri) gerçekleştirebilmesi ve farklı renkler algılandığında karşılık gelen LED ışıklarını gösterebilmesi için parametrelerin nasıl değiştirileceğini belirleyin. Özellikle **Detected Color Twinkle r g b Turn Angle** RGB parametre değerlerini ayarlayarak LED ışıklarının renginin nasıl değiştirildiğine odaklanın.



3 Demo program

