



## OGP101 Oyun Fiziği Final Sınavı — 12 Ocak 2026

Ad Soyad: \_\_\_\_\_ Öğrenci No: \_\_\_\_\_ İmza: \_\_\_\_\_

**⚠ ÖNEMLİ:** Sınav süresi **60 dakikadır**.

### Soru 1

15 puan

Aşağıdaki kod p5.js ve p5.play kütüphanesi kullanılarak yazılmıştır:

```
let g;  
let currentScale = 1;  
let scaleDir = 0.02;  
  
function setup() {  
  new Canvas(400, 300);  
  g = new Sprite(200, 150, 50);  
  g.color = '#c44dff';  
  g.text = '♥';  
  g.textSize = 20;  
}  
  
function draw() {  
  background('#1a1a2e');  
  
  currentScale += scaleDir;  
  if (currentScale > 5 || currentScale < 0.2) {  
    scaleDir *= -1;  
  }  
  g.scale = currentScale;  
}
```

a) Bu kodun canvas üzerinde nasıl bir hareketli görüntü oluşturacağını sözel olarak açıklayınız.

b) `scaleDir *= -1;` satırının işlevini detaylı olarak açıklayınız.

c) `let scaleDir = 0.02;` ifadesi `let scaleDir = 0.1;` olarak değiştirilirse üretilen çıktı nasıl değişir?

Cevap:

### Soru 2

10 puan

Aşağıdaki tabloda bazı fizik kavramları ve p5.play özellik/metodları verilmiştir. Her kavramı doğru özellik/metod ile eşleştiriniz:

| Fizik Kavramı | p5.play Özellik/Metod |
|---------------|-----------------------|
| 1. Hız        | A. sprite.rotation    |
| 2. Sürtünme   | B. sprite.mass        |
| 3. Kütle      | C. sprite.vel         |
| 4. Dönüş      | D. sprite.bounciness  |
| 5. Esneklik   | E. sprite.friction    |

## Soru 3

15 puan

Aşağıdaki kod p5.js ve p5.play kütüphanesi kullanılarak yazılmıştır:

```
1 function setup() {
2   new Canvas(400, 350);
3   world.gravity.y = 10;
4   /* ÖNEMLİ NOT:
5     Konum (200, 340): p5.play'de x,y koordinatları nesnenin MERKEZİNİ temsil eder.
6     x=200 ve genişlik=400 olduğu için, zemin 0'dan 400'e tüm ekranı kaplar.
7     Boyut (400, 20): 3. parametre genişlik, 4. parametre yükseklik belirler.
8     'static': Nesne sabit kalır, yerçekiminden etkilenmez. */
9   let floor = new Sprite(200, 340, 400, 20, 'static');
10  floor.color = '#2d3436';
11
12  for (let i = 0; i < 5; i++) {
13    let ball = new Sprite(80 + i * 70, 50 + i * 30, 35);
14    ball.color = ['white', 'red', 'green', 'blue', 'orange'][i];
15  }
16 }
17
18 function draw() {
19   background('#1a1a2e');
20 }
```

**a)** Bu kod çalıştırıldığında hangi renkli top en erken sabit zemine düşecektir/çarpacaktır? Nedenini açıklayınız.

**b)** Tüm topların aynı anda sabit zemine düşmesi için değiştirmeniz gereken satır numarasını ve bu satırın yeni halini yazınız.

Cevap:

## Soru 4

10 puan

Aşağıdaki p5.js kodlarının her birinin çıktısını yazınız:

**a)**

```
let pos = createVector(5, 3);
let vel = createVector(-2, 4);
pos.add(vel);
print(pos.x, pos.y);
```

**b)**

```
let dir = createVector(6, 8);
print(dir.mag());
```

**c)**

```
let force = createVector(4, 2);
force.mult(3);
print(force.x, force.y);
```

**d)**

```
let start = createVector(10, 5);
let end = createVector(3, 1);
start.sub(end);
start.div(2);
print(start.x, start.y);
```

**e)**

```
let speed = createVector(3, 4);
speed.normalize();
print(speed.mag());
print(speed.x, speed.y);
```

Ad Soyad: \_\_\_\_\_ Öğrenci No: \_\_\_\_\_ İmza: \_\_\_\_\_

## Soru 5

10 puan

p5.js'de renkler hem hexadecimal (hex) hem de RGB formatında tanımlanabilir. Aşağıdaki tabloda her satırdaki iki kod **aynı rengi** üretmektedir. Boşlukları, satırdaki diğer formatın eşdeğeri olacak şekilde doldurunuz:

| Hex Formatı      | RGB Formatı             |
|------------------|-------------------------|
| fill('#3A7D2F'); | fill(____, ____, ____); |
| fill('#_____');  | fill(214, 58, 147);     |

## Soru 6

15 puan

Aşağıdaki p5.js kodu bir "Random Walker" simülasyonu oluşturmaktadır:

```
let walker;

function setup() {
  createCanvas(400, 400);
  background(220);
  walker = new Walker();
}

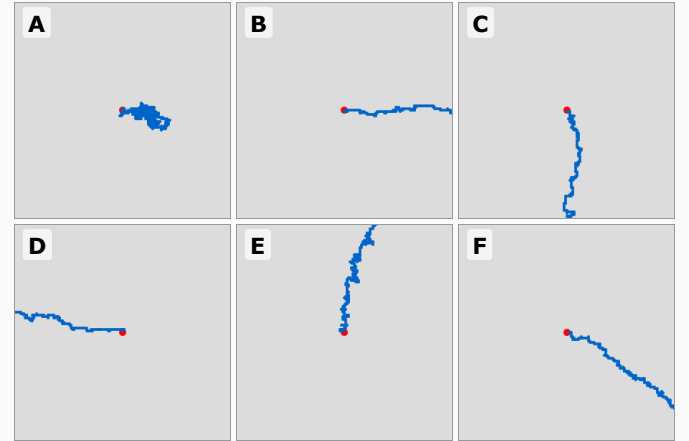
function draw() {
  walker.step();
  walker.show();
}

class Walker {
  constructor() {
    this.x = width / 2;
    this.y = height / 2;
  }

  step() {
    let r = random(1);
    if (r < 0.3) {
      this.x += 1;
    } else if (r < 0.5) {
      this.x -= 1;
    } else if (r < 0.6) {
      this.y += 1;
    } else {
      this.y -= 1;
    }
  }

  show() {
    stroke(0, 100, 200);
    strokeWeight(2);
    point(this.x, this.y);
  }
}
```

Bu kod belirli bir süre çalıştırıldığında, aşağıdaki canvas çıktılarından hangisine benzer bir görüntü oluşturma olasılığı daha yüksektir?



**İstenen:** Doğru seçeneği işaretleyiniz ve yaptığınız seçimin nedenini detaylandırarak açıklayınız.

Cevap:

## Soru 7

15 puan

Bir uzay oyunu yapıyorsunuz. Oyunun arka planında 200 adet yıldız olacak. Yıldızların özellikleri şöyle:

- **Konum (x, y):** Yıldızlar canvas üzerinde (800x600) rastgele herhangi bir yerde olabilir, özel bir yoğunlaşma yok
- **Boyut:** Yıldız boyutları normal dağılım özelliği taşır — ortalama 18 punto, standart sapma 4 punto

**İstenen:** Aşağıdaki `setup()` fonksiyonunun içeri, yukarıdaki özelliklere göre 200 yıldız oluşturacak şekilde tamamlayınız:

```
// textSize(12) → sonraki text() çağrılarını 12 punto boyutunda yapar
// text('★', 100, 200) → (100, 200) konumuna yıldız emoji'si çizer

function setup() {
  createCanvas(800, 600);
  background(0);
  textAlign(CENTER, CENTER);

}
```

## Soru 8

10 puan

Aşağıdaki p5.play kodunda iki kutuya yatay kuvvet uygulanmaktadır. `applyForce(x, y)` fonksiyonunda ilk parametre yatay, ikinci parametre dikey kuvveti belirtir:

```
let kutuA, kutuB;

function setup() {
  new Canvas(600, 300);

  kutuA = new Sprite(150, 150, 50, 50);
  kutuA.mass = 2;
  kutuA.color = 'red';

  kutuB = new Sprite(450, 150, 50, 50);
  kutuB.mass = 8;
  kutuB.color = 'blue';
}

function mousePressed() {
  kutuA.applyForce(100, 0);
  kutuB.applyForce(200, 0);
}
```

**a)** Mouse'a tıklandığında hangi kutu daha fazla ivmelenir?

**b)** Hangi kutunun daha çok ivmeleneceğini bulmak için Newton'un hangi hareket yasasını/formülünü kullanırsınız? Bu formülü kullanarak her iki kutu için ivme hesaplayınız.

Cevap: