

# BURSA TEKNİK ÜNİVERSİTESİ

2021-2022 GÜZ DÖNEMİ

## ALGORİTMALAR VE PROGLAMA PROJE RAPORU

AD SOYAD	ENES BABEKOĞLU
ÖĞRENCİ NUMARASI:	20360859113

### C DİLİNDE TAŞ, KAĞIT, MAKAS OYUNU YAPIMI

#### Belirteçler

Veri Türleri Fonksiyonlar Değişkenler Döngüler break;

#### Ön Bilgilendirme

C programlama dili kullanılarak taş, kağıt, makas oyunu hazırlanırken bilgisayara rastgele 1 - 3 arasında sayı seçtiren ve bunu `int` olarak döndüren bir `bilgisayarınSecimi()` fonksiyonu, oyuncudan cevap alan ve cevabı `int` olarak döndüren bir `oyuncununSecimi()` fonksiyonu, elde edilen puanlara göre kazananın belirlenmesinde ve konsola yazdırılmasında kullanılan `kazananKim()`; `void` fonksiyonu ve oyunun ana fonksiyonu olan `tasKagitMakas()`; `void` fonksiyonu ile bunların C dilinin temel fonksiyonu olan `main()`; `int` fonksiyonu kullanılmıştır. Global değişken tanımlaması yapılmamıştır. Bazı değişkenler main fonksiyonundan bir diğer fonksiyona veya kendi aralarında fonksiyon özellikleri kullanılarak aktarılmıştır. Ayrıca 1 while ve 1 for döngüsü olmak üzere fonksiyonların içerisinde 2 tane döngü kullanılmıştır. Oyun `Switch/Case` yapısı kullanılarak yapılmıştır.

#### Tanımlanan Fonksiyonların Detaylı Tanımlamaları

##### 1 - `int main();`

- ✓ Burada ilk olarak oyunun daha dikkat çekici olması için oyuncunun adını talep ettiğimiz bir değişken olan `isim` değişkenini maksimum 30 karakter olacak şekilde `char` olarak tanımladık.
- ✓ Sonrasında oyuncunun oyunu başlatmak için hazır olup olmadığını sorduğumuz cevabı tutacağımız `int` değerine sahip bir `hazirmi` değişkeni oluşturduk.
- ✓ Daha sonrasında oyunun ayarlarını kolaylıkla değiştirebilmemiz için `int` değerlerine sahip olacak ve bir turda kazanan kişinin alacağı puanı belirten `kazananınAlacağıPuan` ile oyunun bir oyuncu kaç puana ulaşırsa kazanacağını belirten `kacPuandaKazanılacak` şeklinde 2 tane değişken tanımladık. (Bu değerler global olarak sayfanın başına da tanımlanabilirdi.)

✓ Sırasıyla bu fonksiyonu çalıştırdığımızda olacaklar ise şu şekildedir:

1. Bilgisayar, oyuncuya özel bir fonksiyon olan `scanf` ile adını soracaktır ve `isim` değişkenine kaydedecektir.
2. Bilgisayar, oyunun nasıl oynandığını, kazanma durumlarını ve oyunun nasıl kazanılacağını `printf` ile konsola yazarak oyuncuya bildirir ve oyuncuya oyunu başlatmak için hazır olup olmadığını soran ve cevabı kaydeden yeni bir `scanf` fonksiyonu oluşturur. Kullanıcıdan alınan cevabın Evet [1] ve Hayır [2] olması istenmektedir. Bu nedenle bu değerler dışında bir sayı yazıldığında bir döngüyle sorunun tekrar sorulması gerekir. Aksi halde varsa kodlarda bulunan bir sonraki adıma geçilecektir. Bu durumun yapılması için `for(;;){ // burada işlemler yapılıyor }` kodu kullanılarak sonsuz bir döngü oluşturuldu.
  - a) Eğer `scanf` ile oyuncudan alınan `hazirmi` değişkenine Evet [1] cevabı verildiyse oyunun ana fonksiyonu olan `tasKagitMakas()`; fonksiyonu çağırılır. Bu fonksiyonun içerisinde daha sonra kullanabilmek için parantez içinde `isim`, `kazananinAlacagiPuan`, `kacPuandaKazanilacak` değişkenlerini ekledik. (Uyarı: Bunların sırası önemlidir ve fonksiyonun içerisinde aynı sırayla konumlandırılmalıdır.) Sonrasında ise sonsuz döngüden çıkmak için `break`; kalıbını kullandık.
  - b) Eğer `scanf` ile oyuncudan alınan `hazirmi` değişkenine Hayır [2] cevabı verildiyse sonsuz döngüden `break`; kalıbı kullanılarak çıkılır.
  - c) Eğer `scanf` ile oyuncudan alınan `hazirmi` değişkenine 1 ve 2 dışında bir değer girilirse döngü devam edecektir ve soru tekrar sorulacaktır. (Not: Eğer sayı dışında bir değer girilirse döngü tamamen sonsuz olur ve sürekli dönmeye başlayacaktır. Bu tanımladığımız `hazirmi` değişkeninin veri türünün `int` olmasından kaynaklıdır.)

## 2 - void tasKagitMakas();

- ✓ Bu fonksiyon oyunun ana fonksiyonudur. Oyun bu fonksiyonun içerisinde dönen kodlar ile çalışmaktadır. Öncelikle `main()`; fonksiyonunun içerisinde bu fonksiyon çağırılırken gönderilen `isim`, `kazananinAlacagiPuan`, `kacPuandaKazanilacak` değişkenlerine sırasıyla `char isim[30]` (Not: Fonksiyon içerisinde tekrar char tanımlarken karakter sayısını yeniden belirtmeliyiz.), `int kazananinAlacagiPuan` ve `int kacPuandaKazanilacak` şeklinde veri tanımlamalarını yapıyoruz
- ✓ Öncelikle bu fonksiyonun içinde tur sayımızı tutacak olan `i` değişkenini, oyuncumuzun puanını tutacak olan `puanOyuncu` değişkenimizi ve bilgisayarın puanını tutacak olan `puanBilgisayar` değişkenlerini `int` veri tipinde sırasıyla 1, 0 ve 0 olacak şekilde tanımlıyoruz.
- ✓ Sonrasında `while(i){ // burada işlemler yapılıyor i++; }` kodu kullanılarak sonsuz bir döngü oluşturuyoruz. (Not: Bu döngünün sonunda `i++`; kodunu kullanarak `i` değişkeninin her döndürmede 1 artacağını belirttik.) Oyun bu döngünün içerisinde oynanacaktır. Bir oyuncu `kacPuandaKazanilacak` değerine ulaşana kadar döngü devam edecektir. Eğer bir bilgisayarın yada oyuncunun puanı `kacPuandaKazanilacak` değişkeninin değerine ulaşırsa döngü `break`; kodu ile sonlandırılacak ve `kazananKim()`; fonksiyonu konsola bastırılacaktır.

1. Döngünün içerisinde `int` veri tipine sahip `bilgisayarınCevabi` adında bir değişken oluşturuyoruz ve atama operatörü kullanarak `bilgisayarınSecimi()` fonksiyonunu tanımlıyoruz. Bu bize `bilgisayarın rastgele sayı ürettiği` fonksiyonu çağıracaktır.
2. Döngünün içerisinde `int` veri tipine sahip `oyuncununCevabi` adında bir değişken oluşturuyoruz ve atama operatörü kullanarak `oyuncununSecimi()` fonksiyonunu tanımlıyoruz. Bu bize `oyuncunun seçtiği sayıyı döndüren` fonksiyonu çağıracaktır.
3. `Switch/Case` yapısını başlatıyoruz. Öncelikle `switch(oyuncununCevabi)` yazarak oyuncudan alınan cevabın `case:` yapılarıyla kontrol edileceğini bildiriyoruz. Sonrasında 3 tane `case:` yapısı oluşturarak her `case:` yapısı içinde bilgisayarın cevabına göre yanıt veren yeni bir `Switch/Case` yapısı oluşturuyoruz. Bu yapı şu şekilde görünür:

```
➤ switch(oyuncununCevabi){
    case 1: // Eger oyuncu Tas [1] yazarsa:
        switch(bilgisayarınCevabi){
            case 1: // Eger bilgisayar Tas [1] secerse:

                /*
                 * Burada bilgisayarda tas seçme durumunda
                 * yapılacak işlemler yapilir
                 */

                break; // 2. switch 1. case sonu
            .
            .

            default:
                // Burası bilgisayar 1,2,3 disinda bir sayi seçerse
                gorunur

        }

        break; // 1. switch 1. case sonu
    .
    .

    default:
        // 1, 2, 3 disindakiler yazilince goster
}
```

4. Bu döngüde olarak puanların karşılaştırmasını yapıyoruz. Eğer bir kullanıcı `kacPuandaKazanilacak` değerine ulaşmışsa döngüyü `break;` kodu ile sonlandırıyoruz.

5. Bu fonksiyonda son olarak elde ettiğimiz `puanBilgisayar`, `puanOyuncu`, `isim`, `i` ve `kazananinAlacagiPuan` değerlerini `kazananKim()`; fonksiyonunun parantezleri arasında göndererek çağırıyoruz.

### 3 - `void kazananKim();`

- ✓ Bu fonksiyonda öncelikle `tasKagitMakas()`; fonksiyonunun içerisinde bu fonksiyon çağırılırken gönderilen `puanBilgisayar`, `puanOyuncu`, `isim`, `i` ve `kazananinAlacagiPuan` değişkenlerine sırasıyla `int puanBilgisayar`, `int puanOyuncu`, `char isim[30]`, `int i` ve `int kazananinAlacagiPuan` şeklinde veri tanımlamalarını yapıyoruz.
- ✓ Sonrasında ise ekrana yazdıracağımız;
  - Bilgisayarın kaç tur kazandığını hesaplayan `int` veri türünde `bilgisayarTur` adında bir değişken oluşturuyoruz. Bu değişken bilgisayarın aldığı puanı, bir turda kazananın alacağı puana bölerek bilgisayarın kazandığı tur sayısını hesaplar.
  - Oyuncunun kaç tur kazandığını hesaplayan `int` veri türünde `oyuncuTur` adında bir değişken oluşturuyoruz. Bu değişken oyuncunun aldığı puanı, bir turda kazananın alacağı puana bölerek oyuncunun kazandığı tur sayısını hesaplar.
  - Bilgisayar ve oyuncunun kaç turda aynı tahmini yaptığını hesaplayan `int` veri türünde `beraberlikSayisi` adında bir değişken oluşturuyoruz. Bu değişken gelen `i` değişkeninden yani tur sayısından oyuncunun kazandığı tur ile bilgisayarın kazandığı tur sayısının toplamını çıkartarak berabere kalınan tur sayısını hesaplar.
  - Sonrasında ise karşılaştırmalar yapılarak kazanan ve alınan puanlar ile tur sayıları konsola yazdırılır.

### 4 - `int bilgisayarınSecimi();`

- ✓ Bu fonksiyonda öncelikle `srand(time(0));` özel fonksiyonunu çağırarak bilgisayara sistem saati üzerinden rastgele sayı üretmesini söylüyoruz.
- ✓ Sonrasında `int` veri tipinde bir `bilgisayarınCevabi` değişkeni oluşturup bu değişkene rastgele seçilen sayının 1 - 3 arasında olması gerektiğini belirtiyoruz.
- ✓ Rastgele oluşturulan sayının değişkeni olan `bilgisayarınCevabi` değişkenini döndürüyoruz.

### 5 - `int bilgisayarınSecimi();`

- ✓ Gelen tur sayısını `int i` değişkeni olarak parantez içinde tanımlıyoruz. Sonrasında `oyuncununCevabi` değişkeni oluşturuyoruz ve oyuncudan `scanf()`; ile cevap alıp bu değişkene atıyoruz. Son olarak ise `oyuncununCevabi` değişkenini döndürüyoruz.