

# **T.C MANİSA CELAL BAYAR ÜNİVERSİTESİ**

**SEÇİM SİMÜLASYONU YAZILIMI**

**İSMET KIZGIN  
KADİR CAN KARADEMİR**

**DÖNEM PROJESİ  
ALGORİTMA VE PROGRAMLAMA I**

**PROJE DANIŞMANI  
Dr. Öğretim Üyesi Aytuğ ONAN**

**ÖDEV 2**

**MANİSA , 2018**

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

<b>ANALİZ</b>	<b>2</b>
PROBLEM TANIMI	2
Kullanıcıdan İstenilecek Bilgiler	2
Seçim Yapılan İle Özel Çıktı Ekranında Bulunacak Olan Veriler	2
Ülke Geneli Çıktı Ekranında Bulunacak Olan Veriler	3
<b>TASARIM</b>	<b>3</b>
ÇÖZÜM ALGORİTMASI	3
<b>PROGRAMCI KATALOĞU</b>	<b>5</b>
PROJE İÇİN ZAMANLAMA PLANI	5
PROJE KAYNAK ÇIKTI KODU	5
KÜTÜPHANELER	5
GLOBAL DEĞİŞKENLER	6
Seçime Katılan Parti Sayısı Değişkeni	6
PROTOTİPLER	6
MAIN FONKSİYONU	6
Main Fonksiyonu Değişkenleri	6
Parti Sayısı Girdisinin İstenilmesi	7
Oluşturulan Dizi Değişkenlerinin İçlerini Sıfır Atama	7
Do While Döngüsü İçinde Bulunan Kodlar	7
İl Plaka Kodunun Girilmesi	8
Switch Case İşlem Yönlendirmesi	8
FONKSİYONLAR	8
İsimlendirme Fonksiyonu	8
Sıralama Fonksiyonu	9
Milletvekili Dağılımı Tespit Fonksiyonu	10
İl Tanımlama Fonksiyonu	11
Ülke Geneli İstatistik Fonksiyonu	11
Ülke Geneli İstatistik Çıktı Fonksiyonu	12
GitHub KAYNAĞI	12
<b>KULLANICI KATALOĞU</b>	<b>13</b>
Seçime Katılacak Olan Parti Sayısının Girilmesi	13
İle Özel Bilgilerin Girilmesi	13
İle Ait İstatistik Tablosu	14
Ülke Geneli İstatistik Tablosu	14

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### ANALİZ

#### PROBLEM TANIMI

Ülke geneli bir seçimde illere göre oy dağılımı , milletvekili sayısı, iktidar ve ana muhalefet partilerinin belirlenmesi için gerekli seçim simülasyonu yazılımı.

##### Kullanıcıdan İstenilecek Bilgiler

1. Seçime katılacak parti sayısı
2. Seçimin yapıldığı ilin plaka kodu
3. Seçimin yapıldığı ilin milletvekili kontenjanı
4. Seçime katılan partilerin oy miktarlarının girilmesi

Bu bilgiler ışığında partilerin oy oranlarının ve milletvekili dağılımları hesaplanıp, ülke geneli istatistiki veriler elde ediliyor.

##### Seçim Yapılan İle Özel Çıktı Ekranında Bulunacak Olan Veriler

1. İlin plaka kodu
2. Milletvekili kontenjanı
3. İl geneli istatistik tablosu
  - Oy dağılımına göre sıralanması
  - Oy dağılımları
  - Oy oranları
  - Milletvekili dağılımları

Farklı bir il girişi yapılması istenilmektedir. Girilecek illerin bitmesi durumunda ülke geneli istatistiki veriler çıktısı için aşağıdaki bilgiler bulunmalıdır.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### Ülke Geneli Çıktı Ekranında Bulunacak Olan Veriler

1. Ülke geneli milletvekili kontenjanı
2. Ülke geneli geçerli oyların toplamı
3. Ülke geneli istatistik tablosu
  - Partilerin milletvekili sayılarına göre sıralanması
  - Partilerin oy dağılımları
  - Partilerin oy oranları
  - Milletvekili dağılımları
  - Milletvekili oranları
  - İllerde 1. lik durumları
  - İllerden milletvekili alamama durumu
4. İktidar partisi belirtilmesi
5. Ana muhalefet partisinin belirtilmesi

## TASARIM

### ÇÖZÜM ALGORİTMASI

#### Adım 1 Seçime katılacak olan partilerin belirlenmesi:

Ülke geneli seçime girecek olan partilerin sayısının girilmesi. Seçime girilecek olan parti sayısı max 25 olabilir.

#### Adım 2 Seçim yapılan ilin verilerinin girilmesi:

1. İl plaka kodunun girilmesi
2. İle ait milletvekili kontenjanının girilmesi
3. Seçime katılan partilerin oy dağılımlarının girilmesi

Seçim yapılan illerin tümü girildikten sonra 0 plaka kodu ile ülke geneli istatistiki veri alınır.

#### Adım 3 İle özel istatistiki tablo çıktısının oluşturulması:

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

1. İl plaka kodu
2. İlin milletvekili kontenjan sayısı
3. İle ait istatistik tablosu
  1. Partilerin oy dağılımına göre sıralanması
  2. Partilerin oy dağılımı
  3. Partilerin oy oranları
  4. Partilerin milletvekili kontenjan dağılımı

### **Adım 4** Ülke geneli istatistiki verilerinin çıktısının oluşturulması

1. Ülke geneli geçerli toplam oy sayısı
2. Ülke geneli toplam milletvekili kontenjanı
3. Ülke geneli istatistik tablosu
  - Partilerin milletvekili dağılımına göre sıralanması
  - Partilerin oy dağılımı
  - Partilerin oy oranları
  - Partilerin milletvekili kontenjan dağılımı
  - Partilerin milletvekili kontenjan dağılımı oranı
  - Partilerin il 1. lik durumları
  - Partilerin illerden milletvekili çıkmama durumları
4. İktidar partisinin belirtilmesi
5. Ana muhalefet partisinin belirtilmesi

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### PROGRAMCI KATALOĞU

### PROJE İÇİN ZAMANLAMA PLANI

PROJE İÇİN AYRILAN TOPLAM SÜRE: 25.11.2018 - 23.12.2018

SÜREÇ	SÜRE
ANALİZ	20 DAKİKA
TASARIM	20 DAKİKA
GERÇEKLEŞTİRİM	2 SAAT
TEST	20 DAKİKA
RAPORLAMA	4 SAAT

### PROJE KAYNAK ÇIKTI KODU

### KÜTÜPHANELER

```
#include <stdio.h>
#include <stdlib.h>
```

Yazılan kodların çalışması için gerekli kütüphanelerden olan Giriş/Çıkış kütüphanesi stdio.h eklendi ve basit matematiksel işlemler için ise stdlib.h kütüphanesi eklendi.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### GLOBAL DEĞİŞKENLER

#### Seçime Katılan Parti Sayısı Değişkeni

```
int parti_sayisi;
```

Parti sayısının genel bir şekilde fonksiyonlar içinde kullanılabilmesi için global değişken tanımlaması yapıldı.

### PROTOTİPLER

```
void parti_adi_duzenleme(char parti_adi[]);  
void siralama(long long int oy[], char ad[]);  
void mv_sayisi(long long int oy[], int t_kontenjan, int kontenjan[]);  
void il_tanimlama(char parti_adi[], int il_plakasi, char t_parti_adi[], long long int t_oy[], int t_kontenjan[], int birincilik[], int sifir_mv[]);  
void Turkiye_geneli(char parti_adi[], long long int oy[], int kontenjan[], char t_parti_adi[], long long int t_oy[], int t_kontenjan[], int birincilik[], int sifir_mv[]);  
void Turkiye_geneli_cikti(char t_parti_adi[], long long int t_oy[], int t_kontenjan[], int birincilik[], int sifir_mv[]);
```

Kullanılan fonksiyonların main fonksiyon içinde çalışması için gerekli prototiplerin örnekleri.

### MAIN FONKSİYONU

#### Main Fonksiyonu Değişkenleri

```
int il_plakasi;
```

İl plaka kodunun alınabilmesi için gerekli değişkenimiz.

```
char parti_adi[parti_sayisi], t_parti_adi[parti_sayisi];  
int t_kontenjan[parti_sayisi], birincilik[parti_sayisi], sifir_mv[parti_sayisi];  
long long int t_oy[parti_sayisi];
```

Verilerin işlenmesi ve tutulması için gerekli dizilerimizi parti sayısı girildikten sonra oluşturup gerekli işlemlere ataması yapılıyor.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

```
int engel_orani, cocuk_sayisi, buyuk_cocuk_sayisi=0;
float brut_ucret, toplam_brut_ucret, net;
char medeni_durumu, es_durumu, engel_durumu, ad[30], devam;
long int tc;
```

Bilgileri girilen çalışanın bilgilerini almak için oluşturmuş olduğumuz değişkenler.

### Parti Sayısı Girdisinin İstenilmesi

```
while(1)
{
    printf("Lutfen Parti Sayisini Giriniz: ");
    scanf("%d",&parti_sayisi);
    if(parti_sayisi <= 25 && parti_sayisi > 1)
        break;
}
```

Kullanıcıdan parti sayısının girilmesi için gerekli çıktı ekranının verilmesi ve girilen parti sayısı bilgisinin değişkene atanması.

### Oluşturulan Dizi Değişkenlerinin İçlerini Sıfır Atama

```
for(int i=0; i<parti_sayisi; i++)
{
    t_oy[i]=0;
    t_kontenjan[i]=0;
    birincilik[i]=0;
    sifir_mv[i]=0;
}
```

Tutacağımız bilgilerin artışlarını yaparken hata olmushmaması için gerekli atamayı yapıyoruz.

### Do While Döngüsü İçinde Bulunan Kodlar

Sonsuz bir do while döngüsü içine eklenecek olan kodlar.



# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### *İl Plaka Kodunun Girilmesi*

```
printf("\nİlin Plaka Kodunu Giriniz: ");
scanf("%d",&il_plakasi);
```

Kullanıcıdan seçim bilgileri girilecek olan ilin plaka kodunun girilmesi istenmektedir. Çıktı ekranından sonra girilen bilginin değişkene atanması sağlanır.

### *Switch Case İşlem Yönlendirmesi*

```
switch(il_plakasi)
{
    case 0:
        Turkiye_geneli_cikti(t_parti_adi, t_oy, t_kontenjan,birincilik,sifir_mv);
        break;
    default:
        parti_adi_duzenleme(parti_adi);
        il_tanimlama(parti_adi, il_plakasi, t_parti_adi, t_oy, t_kontenjan, birincilik, sifir_mv);
        break;
}
```

Girilen plaka kodunun değerine göre işlem yönlendirmesi yapılır. Gerekli fonksiyonların çalışması sağlanır.

## FONKSİYONLAR

### İsimlendirme Fonksiyonu

```
void parti_adi_duzenleme(char parti_adi[])
{
    char harfler[]={'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','X','Y','Z'};

    for(int i = 0; i<parti_sayisi; i++)
        parti_adi[i]=harfler[i];
}
```

Girilen parti sayısına göre harfler ile partilerin isimlendirmesi sağlanır.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### Sıralama Fonksiyonu

```
void siralama(long long int oy[], char ad[])
{
    long long int gecici_oy;
    char gecici_ad;
    for(int i=0; i<parti_sayisi; i++)
    {
        for(int j=0; j<parti_sayisi-1-i; j++)
        {
            if(oy[j] < oy[j+1])
            {
                gecici_oy = oy[j];
                oy[j] = oy[j+1];
                oy[j+1] = gecici_oy;

                gecici_ad=ad[j];
                ad[j] = ad[j+1];
                ad[j+1] = gecici_ad;
            }
        }
    }
}
```

İllere özel istatistik tablosunda oy dağılımlarına göre sıralama yapılması sağlanır.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### Milletvekili Dağılımı Tespit Fonksiyonu

```
void mv_sayisi(long long int oy[], int t_kontenjan, int kontenjan[])
{
    long long int oylar[parti_sayisi];
    int mv_indis, gecici=0;

    for(int i=0; i<parti_sayisi; i++)
    {
        oylar[i]=oy[i];
        kontenjan[i]=0;
    }

    for(int i=0; i<t_kontenjan; i++)
    {
        for(int j=0; j<parti_sayisi; j++)
        {
            if(oylar[j]>gecici)
            {
                gecici = oylar[j];
                mv_indis = j;
            }
        }
        kontenjan[mv_indis]++;
        oylar[mv_indis] = oylar[mv_indis] / 2;
        gecici=0;
    }
}
```

Parametreler sayesinde alınan oy oranları ve ile özel kontenjan sayısı ile hesaplamalarını gerçekleştiren ve kontenjan dizisine her partinin milletvekili sayısını atanması sağlanır.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

### İl Tanımlama Fonksiyonu

```
void il_tanimlama(char parti_adi[], int il_plakasi, char t_parti_adi[], long long int t_oy[], int t_kontenjan[], int birincilik[], int sifir_mv[])
{
    int m_kontenjan, kontenjan[parti_sayisi];
    long long oy[parti_sayisi], toplam_oy=0;

    printf("İlin Milletvekili Kontenjanini Giriniz: ");
    scanf("%d",&m_kontenjan);
    printf("\n");

    for(int i=0; i<parti_sayisi; i++)
    {
        printf("%c Partisi'nin Oy Sayisini Giriniz: ",parti_adi[i]);
        scanf("%lld",&oy[i]);
        toplam_oy += oy[i];
    }

    printf("\nİl Plaka Kodu: %d\n",il_plakasi);
    printf("Milletvekili Kontenjanı: %d\n",m_kontenjan);
    printf("Geçerli Oy Sayisi: %lld\n",toplam_oy);

    siralama(oy, parti_adi);
    mv_sayisi(oy, m_kontenjan, kontenjan);

    printf("\t\tOy Sayisi\tOy Yuzde\tMV Sayisi\n");
    printf("\t\t-----\t-----\t-----\n");
    for(int i=0; i<parti_sayisi; i++)
        printf("%c Partisi \t\t %lld \t\t %.2f \t\t %d\n",parti_adi[i],oy[i],((float)(oy[i]*100)/toplam_oy),kontenjan[i]);

    Turkiye_geneli(parti_adi, oy, kontenjan, t_parti_adi, t_oy, t_kontenjan, birincilik, sifir_mv);
}
```

İle özel, bilgilerin girilmesini ve çıktı ekranının buluşması için gerekli olan işlemler sağlanır. İle özel işlemler bittikten sonra ülke geneli istatistik hesaplanması için gerekli fonksiyona bilgilerin gönderilmesi sağlanır.

### Ülke Geneli İstatistik Fonksiyonu

```
void Turkiye_geneli(char parti_adi[], long long int oy[], int kontenjan[], char t_parti_adi[], long long int t_oy[], int t_kontenjan[], int birincilik[], int sifir_mv[])
{
    parti_adi_duzenleme(t_parti_adi);

    for(int i=0; i<parti_sayisi; i++)
    {
        for (int j=0; j<parti_sayisi; j++)
        {
            if(t_parti_adi[i] == parti_adi[j])
            {
                t_kontenjan[i]+=kontenjan[j];
                t_oy[i]+=oy[j];
                if(j == 0)
                    birincilik[i]++;
                if(t_kontenjan[i] == 0)
                    sifir_mv[i]++;
                break;
            }
        }
    }
}
```

Parametreler ile almış olduğumuz ülke geneli istatistik değişkenleri ve ile özel bilgilerin tutulduğu değişkenler ile verilerin gerekli işlemlerden geçirilmesi ile istatistik değişkenlerine gerekli atamalar sağlanır.



# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

---

### KULLANICI KATALOĞU

Seçime Katılacak Olan Parti Sayısının Girilmesi

```
Lutfen Parti Sayisini Giriniz: 3
```

Ülke geneli seçime katılacak olan parti sayısının girilmesi.

İle Özel Bilgilerin Girilmesi

```
İlin Plaka Kodunu Giriniz(Baska İl Yok ise 0 Basın): 1
```

İlin plaka kodunun girilmesi istenir. Tüm iller bitmiş ise bu soru sorulduğu zaman 0 kodu girilerek ülke geneli istatistik tablosu alınır.

```
İlin Milletvekili Kontenjanini Giriniz: 3
```

İlin milletvekili kontenjanının girilmesi.

```
A Partisi'nin Oy Sayisini Giriniz: 1567  
B Partisi'nin Oy Sayisini Giriniz: 590  
C Partisi'nin Oy Sayisini Giriniz: 234
```

Seçime katılan tüm partilerin plaka kodu girilen il de aldığı oy miktarlarının tek tek girilmesi.

# ALGORİTMA VE PROGRAMLAMA

## SEÇİM SİMÜLASYONU YAZILIMI

### İle Ait İstatistik Tablosu

```
Il Plaka Kodu: 1
Milletvekili Kontenjani: 3
Geçerli Oy Sayisi: 2391
```

	Oy Sayisi	Oy Yuzde	MV Sayisi
	-----	-----	-----
A Partisi	1567	65.54	2
B Partisi	590	24.68	1
C Partisi	234	9.79	0

İle ait bilgiler girildikten sonra gerekli işlemler yapıp büyükten küçüğe sıralayıp bazı istatistiki veriler verilmektedir.

### Ülke Geneli İstatistik Tablosu

```
Türkiye Geneli
Milletvekili Kontenjani: 3
Geçerli Oy Sayisi: 2391
```

	Oy Sayisi	Oy Yuzdesi	MV Sayisi	MV Yuzdesi	Il 1. lik Sayisi	0 MV Il Sayisi
	-----	-----	-----	-----	-----	-----
A Partisi	1567	65.54	2	66.67	1	0
B Partisi	590	24.68	1	33.33	0	0
C Partisi	234	9.79	0	0.00	0	1

```
Iktidar Partisi: A Partisi
Ana Muhalefet Partisi: B Partisi
```

Tüm iller bittikten sonra gerekli hesaplamalar neticesinde ülke geneli istatistik verileri hesaplanıp milletvekili dağılımlarına göre sıralanıp ekrana basılır.