

ALGORİTMALAR VE PROGRAMLAMA DERSİ

DÖNEM PROJESİ RAPORU

NAZMİ CİRİM

21360859069

Proje Adı:

C programlama dili ile sınav simülatörü

Proje iki kısımdan oluşmaktadır:

1-) Sınav üretici

2-) Sınav değerlendirici

1.KISIM

Sınav Üretici

*Bu kısımda çoktan seçmeli bir test sınavı üretip öğrencilere uygulamak amaçlanmaktadır. Öğrenci sayısı **N** değişkeni ile, soru sayısı **S** değişkeni ile, herhangi bir sorunun boş bırakılma ihtimali **B** değişkeni ile, herhangi bir sorunun doğru cevaplanma ihtimali **D** değişkeni ile kullanıcıdan **scanf** fonksiyonu yardımıyla alınacaktır. Alınan bu değerlere göre bir cevap anahtarı üretilip her bir öğrencinin cevabıyla karşılaştırılacaktır.*

1-) CEVAP ANAHTARI ÜRET

Kullanıcıdan alınan soru sayısı değişkenine göre **time.h** kütüphanesindeki **srand()** ve **rand()** fonksiyonları yardımıyla **cevap_anahtari[]** dizisinin içerisine rastgele bir cevap anahtarı oluşturur.

```
void cevap_anahtari_uret(char cevap_anahtari[], int S){
    int i;
    for(i = 0; i < S; ++i)
        cevap_anahtari[i] = 'A' + (rand() % 5);
}
```

2-) CEVAP ANAHTARI YAZDIR

Oluşturulan cevap anahtari **printf** fonksiyonu yardımıyla **for** döngüsü içerisinde yazdırılır.

```
void cevap_anahtari_yazdir(char cevap_anahtari[], int S){
    int i;
    printf("Cevap Anahtari:\n");
    for(i = 0; i < S; ++i)
        printf("%03d:%c | ", i+1, cevap_anahtari[i]);
    printf("\n");
}
```

3-) SINAVI UYGULA

```
void sinavi_uygula(char ogrenci_cevaplari[][100], char cevap_anahtari[], int N, int S, double B, double D){
    srand(time(NULL));
    int i, j;
    double rastgeleSayi;
    double range = 1;
    double div = RAND_MAX / range;
    for(i = 0; i < N; ++i){
        for(j = 0; j < S; ++j){
            rastgeleSayi = (rand() / div);
            if(rastgeleSayi < B)
                ogrenci_cevaplari[i][j] = 'X';
            else{
                rastgeleSayi = (rand() / div);
                if(rastgeleSayi >= 1.0 - D)
                    ogrenci_cevaplari[i][j] = cevap_anahtari[j];
                else{
                    ogrenci_cevaplari[i][j] = 'A' + (rand() % 5);
                    while(ogrenci_cevaplari[i][j] == cevap_anahtari[j]){
                        ogrenci_cevaplari[i][j] = 'A' + (rand() % 5);
                    }
                }
            }
        }
    }
}
```

4-) ÖĞRENCİ CEVABINI YAZDIR

Öğrencilerin cevapları **ogrenci_ID** ve **for** döngüsü kullanılarak ekrana yazdırılır.

```
void ogrenci_cevabini_yazdir(char ogrenci_cevaplari[][100], int ogrenci_ID, int S){
    int i;
    printf("%03d. ogrencinin cevaplari:\n", ogrenci_ID+1);
    for(i = 0; i < S; ++i){
        printf("%03d:%c | ", i+1, ogrenci_cevaplari[ogrenci_ID][i]);
    }
    printf("\n");
}
```

2.KISIM

Sınav Değerlendirici

Bu kısımda öğrenci cevaplarına göre ham başarı notu, genel sınıf ortalaması, standart sapma, T skoru ve her bir öğrencinin Ti skoruna bakarak aldığı harf notu yazdırılmaktadır.

1-) ÖĞRENCİLERİ PUANLA

*Alınan soru sayısına göre 100 puanlık sınavın her bir sorusunun kaç puan olduğu hesaplanır. Öğrencilerin cevaplarına göre de her bir sorunun puanı, her bir öğrencinin ham başarı notuna eklenir. 4 yanlış 1 doğruyu götürerek şekilde de ham başarı notunda yanlış sayısına göre notu azaltılır. Son olarak her bir öğrencinin ham başarı notu **HBN** dizisi içerisine kaydedilir.*

```
void ogrencileri_puanla(char ogrenci_cevaplari[][100], char cevap_anahtari[], double HBN[], int N, int S){
    int i, j, yanlisSoruSayisi, bosSoru;
    double soruPuanı = 100/S;
    for(i = 0; i < N; ++i){
        bosSoru = 0;
        yanlisSoruSayisi = 0;
        for(j = 0; j < S; ++j){
            if(ogrenci_cevaplari[i][j] == 'X')
                bosSoru++;
            else if(ogrenci_cevaplari[i][j] != cevap_anahtari[j])
                yanlisSoruSayisi++;
        }
        HBN[i] = ( soruPuanı * (S - (yanlisSoruSayisi + bosSoru)) ) - ( soruPuanı * (yanlisSoruSayisi / 4.0) );
        if(HBN[i] < 0)
            HBN[i] = 0;
        //printf("%lf ", HBN[i]);
    }
}
```

2-) SINIF ORTALAMASI HESAPLA

*Her bir öğrencinin ham başarı notuna göre **HBN** dizisi içerisinde bilgiler alınarak sınıfın ortalaması hesaplanır ve ekrana yazdırılır.*

```
double sınıf_ortalama_hesapla(double HBN[], int N){
    double sınıf_ortalama = 0;
    int i;
    for(i = 0; i < N; ++i){
        sınıf_ortalama += HBN[i];
    }
    return (sınıf_ortalama / N);
}
```

3-) STANDART SAPMA HESAPLA

*Math.h kütüphanesinden **pow()** ve **sqrt()** fonksiyonları kullanılarak her bir öğrencinin ham başarı notu ve sınıfın ortalamasına göre standart sapma hesaplanır ve ekrana yazdırılır.*

```
double standart_sapma_hesapla(double ortalama, double HBN[], int N){
    double std, toplam = 0;
    double sayi;
    int i;
    for(i = 0; i < N; ++i){
        sayi = HBN[i] - ortalama;
        toplam = toplam + (pow(sayi, 2));
    }
    toplam = toplam / N;
    double sonuc = sqrt(toplam);
    return sonuc;
}
```

4-) T - SKORU HESAPLA

*Her bir öğrencinin ham başarı notu, sınıf ortalaması ve standart sapmaya göre her bir öğrencinin **T skoru** hesaplanır.*

```
void T_skoru_hesapla(double ortalama, double HBN[], int N, double std, double T_skoru[]){
    int i;
    for(i = 0; i < N; ++i){
        T_skoru[i] = 60 + (10 * ( (HBN[i] - ortalama) / std) );
    }
}
```

5-) HARF NOTU OLUŞTUR

*Her bir öğrencinin ortalaması ve T skoruna göre harf notu verilir.
Her bir öğrencinin harf notu ekrana yazdırılır.*

```
void harf_notu_olustur(double ortalama, double T_skoru[], int N, char harfnotu[][2]){
    int FF, FD1, FD2, DD1, DD2, DC1, DC2, CC1, CC2, CB1, CB2, BB1, BB2, BA1, BA2, AA, i;
    if( ortalama <= 100 ){
        FF = 32;
        FD1 = 32;
        FD2 = 36.99;
        DD1 = 37;
        DD2 = 41.99;
        DC1 = 42;
        DC2 = 46.99;
        CC1 = 47;
        CC2 = 51.99;
        CB1 = 52;
        CB2 = 56.99;
        BB1 = 57;
        BB2 = 61.99;
        BA1 = 62;
        BA2 = 66.99;
        AA = 67;
    }
}
```

```
for(i = 0; i < N; ++i){
    if(T_skoru[i] < FF){
        harfnotu[i][0] = 'F';
        harfnotu[i][1] = 'F';
    }
    else if(FD1 <= T_skoru[i] && T_skoru[i] <= FD2){
        harfnotu[i][0] = 'F';
        harfnotu[i][1] = 'D';
    }
    else if(DD1 <= T_skoru[i] && T_skoru[i] <= DD2){
        harfnotu[i][0] = 'D';
        harfnotu[i][1] = 'D';
    }
    else if(DC1 <= T_skoru[i] && T_skoru[i] <= DC2){
        harfnotu[i][0] = 'D';
        harfnotu[i][1] = 'C';
    }
    else if(CC1 <= T_skoru[i] && T_skoru[i] <= CC2){
        harfnotu[i][0] = 'C';
        harfnotu[i][1] = 'C';
    }
}
```

