**MANİSA**

**CELAL BAYAR ÜNİVERSİTESİ**

**HASAN FERDİ TURGUTLU TEKNOLOJİ FAKÜLTESİ**

ALGORİTMA VE PROGRAMLAMA DERSİ PROJE ÖDEVİ-1 RAPORU

2018-2019

İÇİNDEKİLER

Kapak………………………………………………………….1

İçindekiler…………………………………………………2

Analiz………………………………………………………….3

Tasarım………………………………………………………3-9

Programcı Kataloğu………………………………9

Kullanıcı Kataloğu………………………………….10

Kaynak Kod..……………………………………………..11-20

**#Analiz**

Geliştirilen program bir iş yerinde çalışanların ve iş yerinin kolaylıkla kullanabileceği, bilgilerini hesaplayıp, depolayabileceği bir yazılımdır.

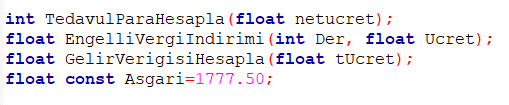
Bizden istenen personel bilgi girişinin alınması ve hesaplanması gereken bilgileri kayıt altına almaktır. İstenildiğinde işlemlerin tekrar edilmesi ya da girilen tüm verilerin yazdırılması gerekmektedir.

**#Tasarım**

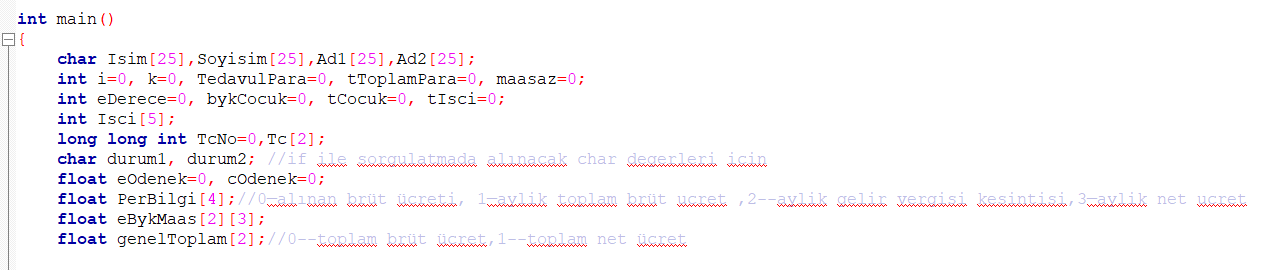
C dilinde yazılmış olup Code Blooks uygulamasında derlenmiştir.

Program içerisinde sabit değişken ve dizilerden yanı sıra, döngüler, fonksiyonlar, karşılaştırma ve koşul deyimlerinden de yararlanılmıştır.

ŞİMDİ KODU İNCELEYELİM



Burada projede kullanmış olduğumuz fonksiyonlar ve sabit değer ifadesi bulunmaktadır.



Burada değişkenlerden bahsedelim biraz:

* Isim ve Soyisim dizilerinde ilk alınan değerleri eşitliyoruz.
* Ad1 ve Ad2 dizileri daha sonra genel toplamda yazdıracağımız isimler için kullanılacaktır.
* İ ve k integer degrleri döngülerimizi döndürmede, Tedavul para hesaplanan tedavül parayı yazdırmada ve tTedavul para ise genel toplamda toplam kaç adet tedavül paranın kullanıldığını aktarmak için kullandık.
* Maasaz değişkenimiz ise maaşı 2000 az olan personel sayısını tutmak için kullanılmıştır.
* eDerece engel derecesi, bykCocuk yaşı 6dan büyük çocuk, tCocuk toplam çocuk sayısını tIsci de toplam isçi sayısını tutmak için tanımlanmıştır.
* Isci[5] dizisi şu şekilde veriler tutmaktadır.

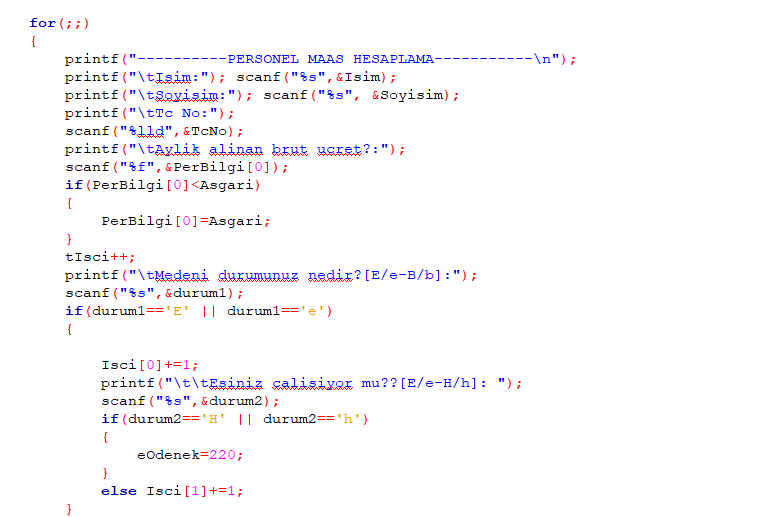
1. Isci[0]—Toplam evli isçi sayısı
2. Isci[1]—Esi çalışan toplam İsci sayisi
3. Isci[2]—Çocuk sayısı
4. Isci[3]—çocuk sayısı 3 ten fazla olan toplam isçi sayısı
5. Isci[4]—toplam engelli isçi sayısı

* eOdenek ve cOdenek eş ve çocuk için alınan para miktarı
* eBuyukMaas[2][3] programın bizden en son olarak istediği değişkenleri atar

1. eBykMaas[0][0]--en yüksek aylık top brüt ücret
2. eBykMaas[0][1]--gelir vergisi
3. eBykMaas[0][2]--Net ücret

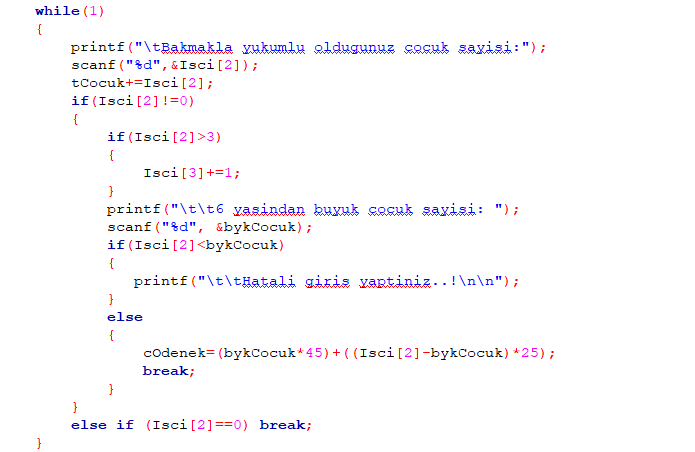
* eBykMaas[1][2]--en yüksek aylık top brüt ücret

1. eBykMaas[1][1]--gelir vergisi
2. eBykMaas[1][2]--Net ücret



Burada ilk veri girişleri bulunmaktadır.

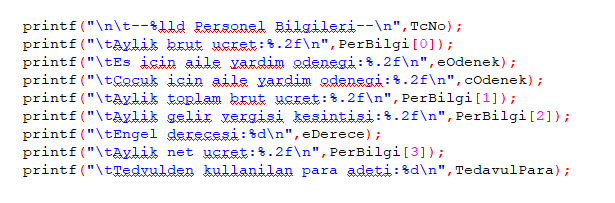
Medeni durum sorulup eOdenek eklenmektedir.



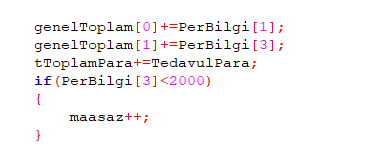
Çocuk sayısı alınmaktadır. cOdenek e çocuk sayısı kadar ödenek eklenmiştir.



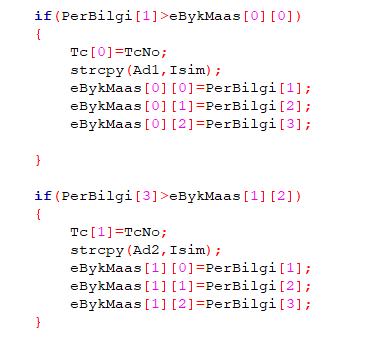
Burada da çalışanımızın sağlık durumu girilmesi istenip belirtilen değerlerin sorgulaması yapılmaktadır.



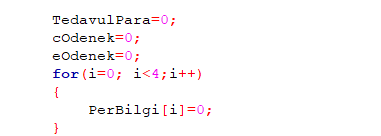
Son olarak bilgileri alınan çalışanın verileri ekranda görüntülenmektedir.



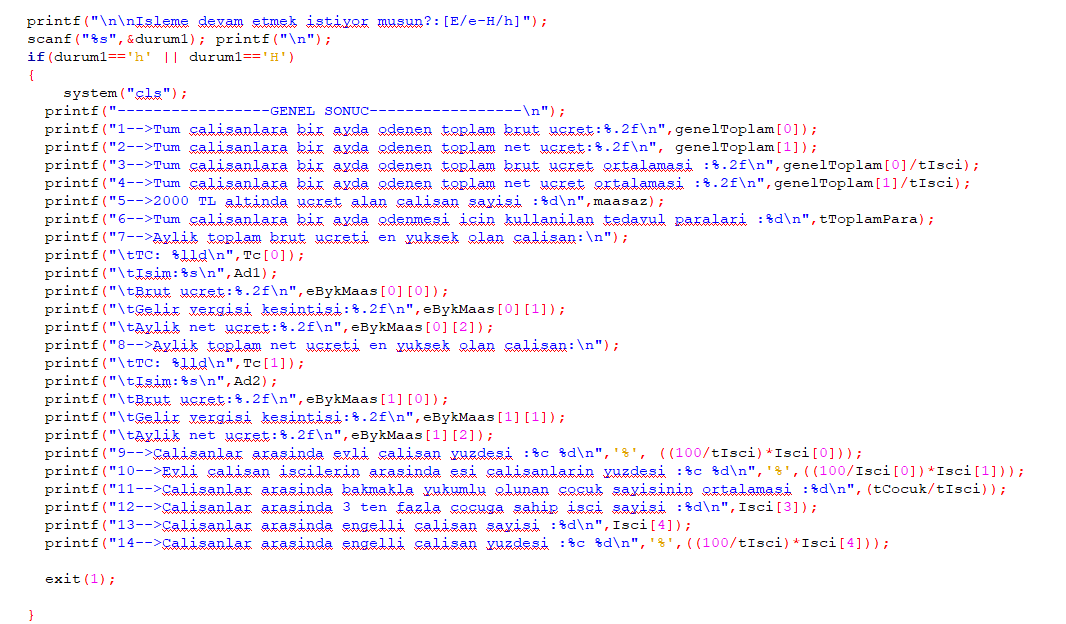
Burada genel toplamda yazdırmamız istenilen dizilerin içi doldurulmaktadır.



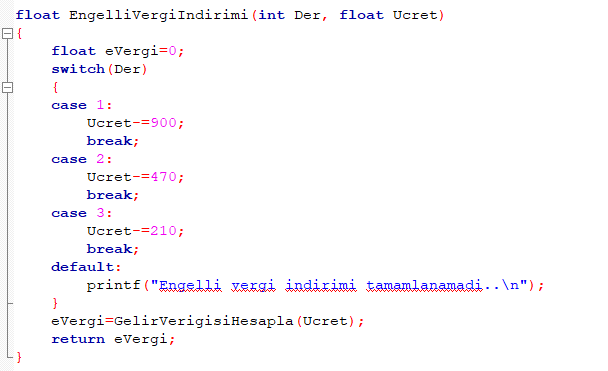
Bizden istenilen en büyük brüt ve en büyük net ücretli çalışan için her işçi bilgi girdiğinde sorgulatma yapıp deger atanır.

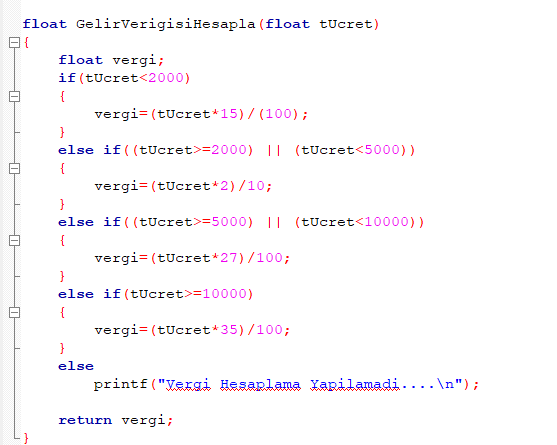


Yeni işçi girişi için sürekli kulandığımız verilerin içi sıfırlanır.

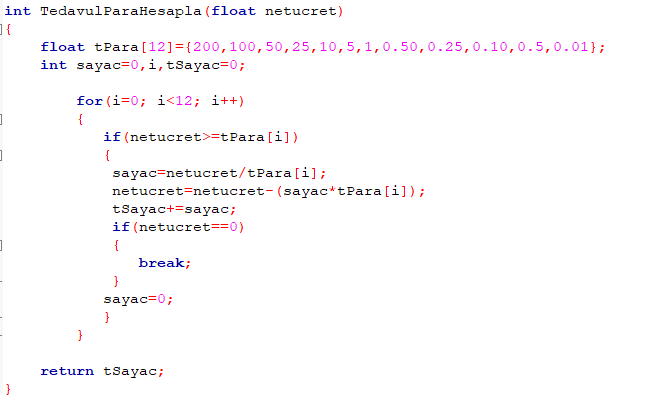


işleme devam etmek isterse kullanıcı en baştan personel bilgileri almaya devam edecektir. Hayır derse genel toplam çıktısı karşısına çıkacaktır.



Bu kodda engelli çalışanlar için uygulanacak vergi indirimi görülmektedir.

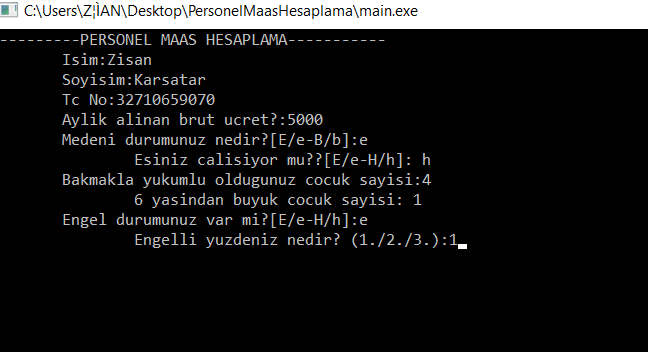
Bu fonksiyonumuzda da vergi indirimi hesabımız yapılmaktadır.



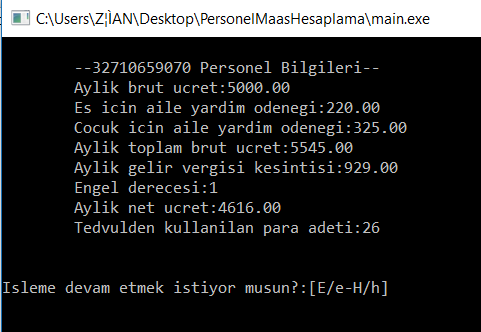
Son olarak tedavülden kullandığımız paraları hesaplamak için yararlandığımız fonksiyonumuz.

#Programcı Kataloğu

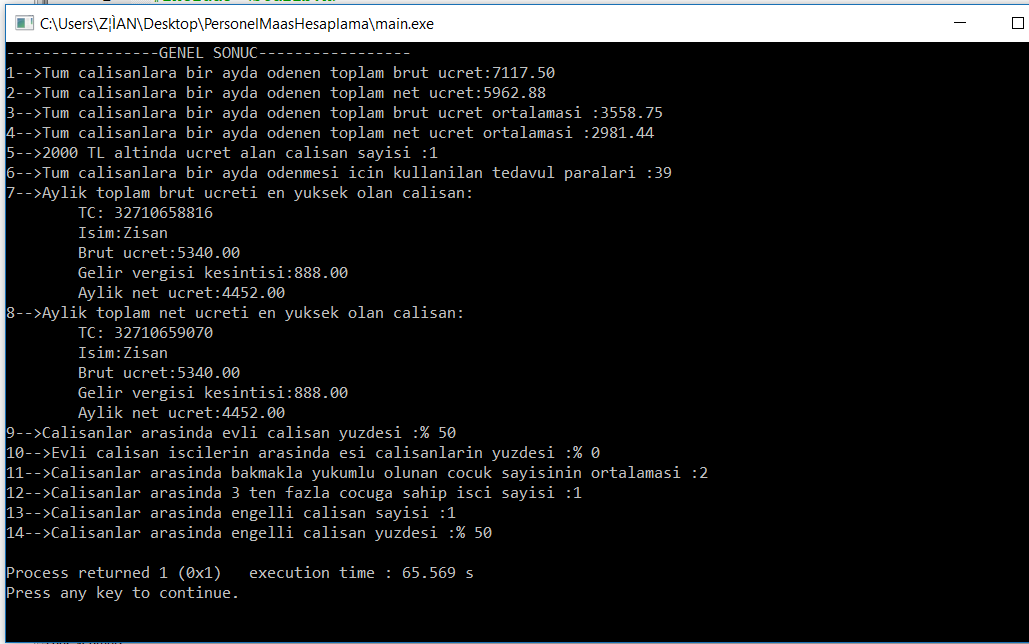
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1. Hafta | 2.Hafta | 3.Hafta | 4.Hafta |
| Açıklama | Analiz ve Tasarım | Kodlama | - | Test ve Raporlama |
| İsmail Bakır | 2 gün arayla 1 saat | Her gün 1 saat | - | Günde yarım saat |
| Zişan Karsatar | 2 gün arayla 1 saat | Her gün 1 saat | - | Günde yarım saat |

****#Kullanıcı Kataloğu

Burada ilk olarak personel bilgileri alınmaktadır.

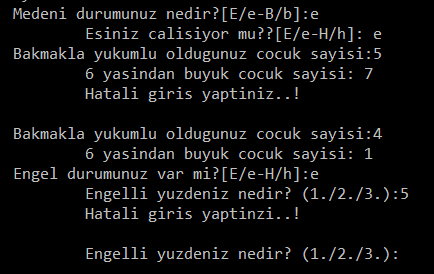


Burada da girilen personelin bilgileri ekrana basılmaktadır. Her yeni personel için gerçekleşen bir işlemdir.



Bu ekran çıktısında girilen tüm veriler ekrana bastırılmıştır. Ve program kapanır.

KISITLAMALAR



* Medeni durumunuza E/e cevabı verdiğinizde eşinizin çalışıp çalışmadığı merak edilmektedir. Eşi çalışmayan işçiye eş ödeneği sağlanmaktadır. B/b cevabı verildiğinde ise eşiniz çalışıyor mu sorusu atlanmaktadır.
* Bakmakla yükümlü olunan çocuk sayısı alınıp başka soruya yönlendirilmekte. Eğer ikinci soruya daha büyük bir değer girilirse program hata mesajını yayınlayıp tekrar giriş yapılması istenilmekte. Aksi takdirde çocuk sayısı 0 girildiğinde bu soru atlanılmakta.
* Engel durumu alınan işçi E/e cevabı verdiyse engel derecesi alınmakta ve uygun koşulların girilmesi sağlanmaktadır.

#Kaynak Kod

#include <stdio.h>

#include <stdlib.h>

int TedavulParaHesapla(float netucret);

float EngelliVergiIndirimi(int Der, float Ucret);

float GelirVerigisiHesapla(float tUcret);

float const Asgari=1777.50;

int main()

{

char Isim[25],Soyisim[25],Ad1[25],Ad2[25];

int i=0, k=0, TedavulPara=0, tToplamPara=0, maasaz=0;

int eDerece=0, bykCocuk=0, tCocuk=0, tIsci=0;

int Isci[5];

long long int TcNo=0,Tc[2];

char durum1, durum2; //if ile sorgulatmada alýnacak char degerleri icin

float eOdenek=0, cOdenek=0;

float PerBilgi[4];//0—alýnan brüt ücreti, 1—aylik toplam brüt ucret ,2--aylik gelir vergisi kesintisi,3—aylik net ucret

float eBykMaas[2][3];

float genelToplam[2];//0--toplam brüt ücret,1--toplam net ücret

for(i=0;i<5;i++)

{

Isci[i]=0;

}

for(k=0; k<2; k++)

{

for(i=0; i<3;i++)

{

eBykMaas[k][i]=0;

}

}

for(;;)

{

printf("----------PERSONEL MAAS HESAPLAMA-----------\n");

printf("\tIsim:"); scanf("%s",&Isim);

printf("\tSoyisim:"); scanf("%s", &Soyisim);

printf("\tTc No:");

scanf("%lld",&TcNo);

printf("\tAylik alinan brut ucret?:");

scanf("%f",&PerBilgi[0]);

if(PerBilgi[0]<Asgari)

{

PerBilgi[0]=Asgari;

}

tIsci++;

printf("\tMedeni durumunuz nedir?[E/e-B/b]:");

scanf("%s",&durum1);

if(durum1=='E' || durum1=='e')

{

Isci[0]+=1;

printf("\t\tEsiniz calisiyor mu??[E/e-H/h]: ");

scanf("%s",&durum2);

if(durum2=='H' || durum2=='h')

{

eOdenek=220;

}

else Isci[1]+=1;

}

while(1)

{

printf("\tBakmakla yukumlu oldugunuz cocuk sayisi:");

scanf("%d",&Isci[2]);

tCocuk+=Isci[2];

if(Isci[2]!=0)

{

if(Isci[2]>3)

{

Isci[3]+=1;

}

printf("\t\t6 yasindan buyuk cocuk sayisi: ");

scanf("%d", &bykCocuk);

if(Isci[2]<bykCocuk)

{

printf("\t\tHatali giris yaptiniz..!\n\n");

}

else

{

cOdenek=(bykCocuk\*45)+((Isci[2]-bykCocuk)\*25);

break;

}

}

else if (Isci[2]==0) break;

}

PerBilgi[1]=PerBilgi[0]+cOdenek+eOdenek;

printf("\tEngel durumunuz var mi?[E/e-H/h]:");

scanf("%s",&durum1);

if(durum1=='e' || durum1=='E')

{

while(1)

{

printf("\t\tEngelli yuzdeniz nedir? (1./2./3.):");

scanf("%d",&eDerece);

if(eDerece<=0 || eDerece>3)

{

printf("\t\tHatali giris yaptinzi..!\n\n");

}

else

{

Isci[4]+=1;

PerBilgi[2]=EngelliVergiIndirimi(eDerece,PerBilgi[1]);

break;

}

}

}

else if(durum1=='h' || durum1=='H')

{

PerBilgi[2]=GelirVerigisiHesapla(PerBilgi[1]);

}

else printf("\t\tGirdiniz anlasilmamistir..\n\n");

PerBilgi[3]=PerBilgi[1]-PerBilgi[2];

TedavulPara=TedavulParaHesapla(PerBilgi[3]);

system("cls");

printf("\n\t--%lld Personel Bilgileri--\n",TcNo);

printf("\tAylik brut ucret:%.2f\n",PerBilgi[0]);

printf("\tEs icin aile yardim odenegi:%.2f\n",eOdenek);

printf("\tCocuk icin aile yardim odenegi:%.2f\n",cOdenek);

printf("\tAylik toplam brut ucret:%.2f\n",PerBilgi[1]);

printf("\tAylik gelir vergisi kesintisi:%.2f\n",PerBilgi[2]);

printf("\tEngel derecesi:%d\n",eDerece);

printf("\tAylik net ucret:%.2f\n",PerBilgi[3]);

printf("\tTedvulden kullanilan para adeti:%d\n",TedavulPara);

genelToplam[0]+=PerBilgi[1];

genelToplam[1]+=PerBilgi[3];

tToplamPara+=TedavulPara;

if(PerBilgi[3]<2000)

{

maasaz++;

}

if(PerBilgi[1]>eBykMaas[0][0])

{

Tc[0]=TcNo;

strcpy(Ad1,Isim);

eBykMaas[0][0]=PerBilgi[1];

eBykMaas[0][1]=PerBilgi[2];

eBykMaas[0][2]=PerBilgi[3];

}

if(PerBilgi[3]>eBykMaas[1][2])

{

Tc[1]=TcNo;

strcpy(Ad2,Isim);

eBykMaas[1][0]=PerBilgi[1];

eBykMaas[1][1]=PerBilgi[2];

eBykMaas[1][2]=PerBilgi[3];

}

TedavulPara=0;

cOdenek=0;

eOdenek=0;

for(i=0; i<4;i++)

{

PerBilgi[i]=0;

}

printf("\n\nIsleme devam etmek istiyor musun?:[E/e-H/h]");

scanf("%s",&durum1); printf("\n");

if(durum1=='h' || durum1=='H')

{

system("cls");

printf("-----------------GENEL SONUC-----------------\n");

printf("1-->Tum calisanlara bir ayda odenen toplam brut ucret:%.2f\n",genelToplam[0]);

printf("2-->Tum calisanlara bir ayda odenen toplam net ucret:%.2f\n", genelToplam[1]);

printf("3-->Tum calisanlara bir ayda odenen toplam brut ucret ortalamasi :%.2f\n",genelToplam[0]/tIsci);

printf("4-->Tum calisanlara bir ayda odenen toplam net ucret ortalamasi :%.2f\n",genelToplam[1]/tIsci);

printf("5-->2000 TL altinda ucret alan calisan sayisi :%d\n",maasaz);

printf("6-->Tum calisanlara bir ayda odenmesi icin kullanilan tedavul paralari :%d\n",tToplamPara);

printf("7-->Aylik toplam brut ucreti en yuksek olan calisan:\n");

printf("\tTC: %lld\n",Tc[0]);

printf("\tIsim:%s\n",Ad1);

printf("\tBrut ucret:%.2f\n",eBykMaas[0][0]);

printf("\tGelir vergisi kesintisi:%.2f\n",eBykMaas[0][1]);

printf("\tAylik net ucret:%.2f\n",eBykMaas[0][2]);

printf("8-->Aylik toplam net ucreti en yuksek olan calisan:\n");

printf("\tTC: %lld\n",Tc[1]);

printf("\tIsim:%s\n",Ad2);

printf("\tBrut ucret:%.2f\n",eBykMaas[1][0]);

printf("\tGelir vergisi kesintisi:%.2f\n",eBykMaas[1][1]);

printf("\tAylik net ucret:%.2f\n",eBykMaas[1][2]);

printf("9-->Calisanlar arasinda evli calisan yuzdesi :%c %d\n",'%', ((100/tIsci)\*Isci[0]));

printf("10-->Evli calisan iscilerin arasinda esi calisanlarin yuzdesi :%c %d\n",'%',((100/Isci[0])\*Isci[1]));

printf("11-->Calisanlar arasinda bakmakla yukumlu olunan cocuk sayisinin ortalamasi :%d\n",(tCocuk/tIsci));

printf("12-->Calisanlar arasinda 3 ten fazla cocuga sahip isci sayisi :%d\n",Isci[3]);

printf("13-->Calisanlar arasinda engelli calisan sayisi :%d\n",Isci[4]);

printf("14-->Calisanlar arasinda engelli calisan yuzdesi :%c %d\n",'%',((100/tIsci)\*Isci[4]));

exit(1);

}

}

return 0;

}

float EngelliVergiIndirimi(int Der, float Ucret)

{

float eVergi=0;

switch(Der)

{

case 1:

Ucret-=900;

break;

case 2:

Ucret-=470;

break;

case 3:

Ucret-=210;

break;

default:

printf("Engelli vergi indirimi tamamlanamadi..\n");

}

eVergi=GelirVerigisiHesapla(Ucret);

return eVergi;

}

float GelirVerigisiHesapla(float tUcret)

{

float vergi;

if(tUcret<2000)

{

vergi=(tUcret\*15)/(100);

}

else if((tUcret>=2000) || (tUcret<5000))

{

vergi=(tUcret\*2)/10;

}

else if((tUcret>=5000) || (tUcret<10000))

{

vergi=(tUcret\*27)/100;

}

else if(tUcret>=10000)

{

vergi=(tUcret\*35)/100;

}

else

printf("Vergi Hesaplama Yapilamadi....\n");

return vergi;

}

int TedavulParaHesapla(float netucret)

{

float tPara[12]={200,100,50,25,10,5,1,0.50,0.25,0.10,0.5,0.01};

int sayac=0,i,tSayac=0;

for(i=0; i<12; i++)

{

if(netucret>=tPara[i])

{

sayac=netucret/tPara[i];

netucret=netucret-(sayac\*tPara[i]);

tSayac+=sayac;

if(netucret==0)

{

break;

}

sayac=0;

}

}

return tSayac;

}