

ÖĞRENCİ İSMİ:

SÜRENİZ 10 DAKİKADIR.

DERS NO(3 basamaklı):

BİL141 KISA SINAV V ŞUBE 1 GRUP J SORULARI

- ASİSTANLARIN SINAV ESNASINDAKİ TEK GÖREVİ SINIFI GÖZETLEMektir. SORULARDAN PUAN ALABİLMENİZ İÇİN TAM OLARAK CEVAP VERMENİZ GEREKMEKTEDİR. KISMI PUANLAMA YAPILMAMAKTADIR. Örneğin, cevap 14.3 ise 14.30 yazan öğrenci o sorudan puan alamaz! / ve \ farkı, ' ve " farkı, $\leq$ yerine $\leq$ gibi detaylar da dâhil olmak üzere CEVABINIZIN TAM OLARAK DOĞRULUĞUNA emin olunuz. Ancak .(nokta) yerine ,(virgül) koymak ya da C yerine c yazmak gibi durumlar, soruda özellikle ölçülenler arasında değilse, puan kaybına yol açmaz.
- EKRAN ÇIKTISI SORULARINDA KODLARIN ÇIKTISINI HER HÜCRE BİR KARAKTERİ TEMSİL EDECEK ŞEKİLDE KUTULARA YAZINIZ. Cevabınızın kutuya sığmadığı durumlar olursa, ilgili kısma ek hücreler çizerek cevabınızı belirtebilirsiniz. Hata içerdiğini düşündüğünüz sorularda (mantık hatası haricindekilerde) ne tür bir hata olduğunu aşağıdaki seçeneklerden seçerek cevap kutusunun içine (örn. "HATA1" şeklinde) yazınız VE nedenini cevap kutusunun altına ya da soruya yakın başka boş bir alana (örn. "Çünkü koddaki ... eksik." şeklinde) belirtiniz. HATA1:Kod hiç derlenemeyecektir. HATA2:Kod derlenecek ama çalışması esnasında hata verecektir. HATA3:Kod derlenecek ama çalışması esnasında sonsuz döngüye girecektir. Yukarıdaki ifadelere uymayan hatalarda (örneğin mantık hatası içerdiğini düşündüğünüz sorularda) hatasız sorularda olduğu gibi sadece ekran çıktısını yazınız, HATA# vs. yazmayınız.
- KODDA BOŞLUK SORULARINDA EKRAN ÇIKTISINI SAĞLAYACAK ŞEKİLDE BOŞLUĞU KULLANDIĞINIZ TÜM KARAKTERLERE ($\leq$ yerine $\leq$ gibi) DİKKAT EDEREK DOLDURUNUZ. Gereksiz komut/ifadeler eklemeyiniz; bu tür durumlarda kodunuz çıktı açısından isteneni yapıyor olsa bile puan alamayabilirsiniz.

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

```
typedef struct
```

```
{  
    int x,y[3];
```

```
} var;
```

```
void fonk(var*);
```

```
int main()
```

```
{  
    var yok;  
    yok.x=7;  
    yok.y[2]=yok.x;  
    printf("%d%d\n",yok.x,yok.y[2]);  
    fonk(&yok);  
    printf("%d%d\n",yok.x,yok.y[2]);  
    return 0;  
}
```

```
void fonk(var* P)
```

```
{  
    if(P->x<-9)  
        P->y[2]=4;  
    (*P).x=2;  
}
```

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

```
#include <string.h>
```

```
typedef union // union(soğan) değil!
```

```
{  
    char sirinlsim[12];
```

```
    int boy;
```

```
} smurf;
```

```
int main()
```

```
{  
    smurf gozluklu;  
    gozluklu.boy=15;  
    printf("%d",gozluklu.boy);  
    strcpy(gozluklu.sirinlsim, "Uykucu");  
    if(gozluklu.boy+1 == 16)  
        printf("%d\n", gozluklu.boy+17);  
    printf("%d", (int)strlen(gozluklu.sirinlsim));  
    return 0;  
}
```

1	5	6							

7	7								
2	7								

KODDA BOŞLUK DOLDURMA SORUSU

```
#include <stdio.h>

void faktoriyelMi( int, int*);

int main()
{
    int sayi;
    int k;
    printf("Sayi:");
    scanf("%d", &sayi);
    faktoriyelMi(sayi, &k);
    if(k!=-1)
        printf("%d = %d!",sayi, k);
    else
        printf("%d sayisi hicbir sayinin faktoriyeli degildir.",sayi);
    return 0;
}

void faktoriyelMi( int sayi, int* h )
{
    int b=2;
    while (sayi%b==0 && sayi!=b)
        sayi/=b++ ;

    *h = (sayi == b)?b:-1;
}
```

ÖRNEK EKRAN ÇIKTISI

Sayi:2 2 = 2! ----- Process exited after 1.3 Devam etmek için bir tuş	Sayi:120 120 = 5! ----- Process exited after 5.6 Devam etmek için bir tuş
---	---

Soldaki çıktıda kod çalıştırdıktan sonra kullanıcı "2" girişi yapmış ve enter'a basmıştır. 2 sayısı, 1 ve 2 sayılarının çarpımı olduğu için 2! dir. Sağdaki çıktıda da kod çalıştırıldıktan sonra kullanıcı "120" girişi yapmış ve enter'a basmıştır.

ÖRNEK EKRAN ÇIKTISI

Sayi:3 3 sayisi hicbir sayinin faktoriyeli degildir. ----- Process exited after 1.3 Devam etmek için bir tuş
--

Kod çalıştırıldıktan sonra kullanıcı "3" girişi yapmış ve enter'a basmıştır. 3 sayısı bir sayının faktöriyeli olmadığı için belirtilen ifade ekrana yazdırılmıştır.

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

```
#include <string.h>
```

```
typedef struct
```

```
{
    char isim[10];

    int fiyat;
} apple;

int main()
{
    apple elma = {"sari", 5};

    // a'nin ASCII'si 97'dir.

    strcpy(elma.isim, "98");

    printf("%s", elma.isim);

    printf("%d", elma.fiyat);

    return 0;
}
```

9	8	5							

BU KISA SINAVDA 2 SORUDAKİ KODLAR DEVCPP'TAKİ GİBİ TONLAMALI OLARAK BASTIRILMIŞTIR. DÖNEM SONU SINAVINDA DA KODLAR BU ŞEKİLDE BASTIRILABİLİR. TERCİHİNİZİ İŞARETLEYİNİZ VE DİLERSENİZ YORUMLARINIZI EKLEYİNİZ:

a) (Bundan önce olduğu gibi) bütün karakterlerin aynı tonda olmasını tercih ederim.

b) Tonlamalı (bazı karakterler **koyu**, bazıları *gri*...) olmasını tercih ederim:

c) Benim için ikisi de olur. Fark etmez.

Yorumunuz:

ÖĞRENCİ İSMİ:

SÜRENİZ 10 DAKİKADIR.

DERS NO(3 basamaklı):

BİL141 KISA SINAV V ŞUBE 1 GRUP K SORULARI

- ASİSTANLARIN SINAV ESNASINDAKİ TEK GÖREVİ SINIFI GÖZETLEMektir. SORULARDAN PUAN ALABİLMENİZ İÇİN **TAM OLARAK CEVAP VERMENİZ** GEREKMEKTEDİR. KİSMİ PUANLAMA YAPILMAMAKTADIR. Örneğin, cevap 14.3 ise 14.30 yazan öğrenci o sorudan puan alamaz! / ve \ farkı, ' ve " farkı, $\leq$ yerine $\leq$ gibi detaylar da dâhil olmak üzere **CEVABINIZIN TAM OLARAK DOĞRULUĞUNA** emin olunuz. Ancak .(nokta) yerine ,(virgül) koymak ya da C yerine c yazmak gibi durumlar, soruda özellikle ölçülenler arasında değilse, puan kaybına yol açmaz.
- EKRAN ÇIKTISI SORULARINDA KODLARIN ÇIKTISINI HER HÜCRE BİR KARAKTERİ TEMSİL EDECEK ŞEKİLDE KUTULARA YAZINIZ. Cevabınızın kutuya sığmadığı durumlar olursa, ilgili kısma ek hücreler çizerek cevabınızı belirtebilirsiniz. Hata içerdiğini düşündüğünüz sorularda (mantık hatası haricindekilerde) ne tür bir hata olduğunu aşağıdaki seçeneklerden seçerek cevap kutusunun içine (örn. "HATA1" şeklinde) yazınız **VE** nedenini cevap kutusunun altına ya da soruya yakın başka boş bir alana (örn. "Çünkü koddaki ... eksik." şeklinde) belirtiniz. HATA1:Kod hiç derlenemeyecektir. HATA2:Kod derlenecek ama çalışması esnasında hata verecektir. HATA3:Kod derlenecek ama çalışması esnasında sonsuz döngüye girecektir. Yukarıdaki ifadelere uymayan hatalarda (örneğin mantık hatası içerdiğini düşündüğünüz sorularda) hatasız sorularda olduğu gibi sadece ekran çıktısını yazınız, HATA# vs. yazmayınız.
- KODDA BOŞLUK SORULARINDA EKRAN ÇIKTISINI SAĞLAYACAK ŞEKİLDE BOŞLUĞU KULLANDIĞINIZ TÜM KARAKTERLERE ($\leq$ yerine $\leq$ gibi) DİKKAT EDEREK DOLDURUNUZ. Gereksiz komut/ifadeler eklemeyiniz; bu tür durumlarda kodunuz çıktı açısından isteneni yapıyor olsa bile puan alamayabilirsiniz.

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

```
#include <string.h>
```

```
typedef union // onion(soğan) değil!
```

```
{
```

```
    char sirinlsim[14];
```

```
    int boy;
```

```
} smurf;
```

```
int main()
```

```
{
```

```
    smurf gozluklu;
```

```
    gozluklu.boy=22;
```

```
    printf("%d",gozluklu.boy);
```

```
    strcpy(gozluklu.sirinlsim, "Somurtkan");
```

```
    if(gozluklu.boy+1 == 23)
```

```
        printf("%d\n", gozluklu.boy+24);
```

```
    printf("%d", (int)strlen(gozluklu.sirinlsim));
```

```
    return 0;
```

```
}
```

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

```
#include <string.h>
```

```
typedef struct
```

```
{
```

```
    char isim[10];
```

```
    int fiyat;
```

```
} apple;
```

```
int main()
```

```
{
```

```
    apple elma = {"sari", 7};
```

```
    // a'nin ASCII'si 97'dir.
```

```
    strcpy(elma.isim, "99");
```

```
    printf("%s", elma.isim);
```

```
    printf("%d", elma.fiyat);
```

```
    return 0;
```

```
}
```

2	2	9												

9	9	7												

KODDA BOŞLUK DOLDURMA SORUSU

```
#include <stdio.h>

void faktoriyelMi( int, int*);

int main()
{
    int sayi;
    int k;
    printf("Sayi:");
    scanf("%d", &sayi);
    faktoriyelMi(sayi, &k);
    if(k!=-1)
        printf("%d = %d!",sayi, k);
    else
        printf("%d sayisi hicbir sayinin faktoriyeli degildir.",sayi);
    return 0;
}

void faktoriyelMi( int sayi, int*w )
{
    int b=2;
    while (sayi%b==0 && sayi!=b)
        sayi/=b++;

    *w = (sayi == b)?b:-1;
}
```

ÖRNEK EKRAN ÇIKTISI

Sayi:2 2 = 2! ----- Process exited after 1.3 Devam etmek için bir tuş	Sayi:120 120 = 5! ----- Process exited after 5.6 Devam etmek için bir tuş
---	---

Soldaki çıktıda kod çalıştırıldıktan sonra kullanıcı "2" girişi yapmış ve enter'a basmıştır. 2 sayısı, 1 ve 2 sayılarının çarpımı olduğu için 2!dir. Sağdaki çıktıda da kod çalıştırıldıktan sonra kullanıcı "120" girişi yapmış ve enter'a basmıştır.

ÖRNEK EKRAN ÇIKTISI

Sayi:3 3 sayisi hicbir sayinin faktoriyeli degildir. ----- Process exited after 1.3 Devam etmek için bir tuş

Kod çalıştırıldıktan sonra kullanıcı "3" girişi yapmış ve enter'a basmıştır. 3 sayısı bir sayının faktöriyeli olmadığı için belirtilen ifade ekrana yazdırılmıştır.

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

```
typedef struct
```

```
{
    int x,y[3];
}
```

```
var;
```

```
void fonk(var*);
```

```
int main()
```

```
{
    var yok;
    yok.x=6;
    yok.y[2]=yok.x;
    printf("%d%d\n",yok.x,yok.y[2]);
    fonk(&yok);
    printf("%d%d\n",yok.x,yok.y[2]);
    return 0;
}
```

```
void fonk(var* P)
```

```
{
    if(P->x<-9)
        P->y[2]=4;
    (*P).x=8;
}
```

6	6								
8	6								

BU KISA SINAVDA 2 SORUDAKİ KODLAR DEVCPP'TAKİ GİBİ TONLAMALI OLARAK BASTIRILMIŞTIR. DÖNEM SONU SINAVINDA DA KODLAR BU ŞEKİLDE BASTIRILABİLİR. TERCİHİNİZİ İŞARETLEYİNİZ VE DİLERSENİZ YORUMLARINIZI EKLEYİNİZ:

a) (Bundan önce olduğu gibi) bütün karakterlerin aynı tonda olmasını tercih ederim.

b) Tonlamalı (bazı karakterler **koyu**, bazıları *gri*...) olmasını tercih ederim:

c) Benim için ikisi de olur. Fark etmez.

Yorumunuz:

ÖĞRENCİ İSMİ:

SÜRENİZ 10 DAKİKADIR.

DERS NO(3 basamaklı):

BİL141 KISA SINAV V ŞUBE 2 GRUP L SORULARI

- ASİSTANLARIN SINAV ESNASINDAKİ TEK GÖREVİ SINIFI GÖZETLEMektir. SORULARDAN PUAN ALABİLMENİZ İÇİN TAM OLARAK CEVAP VERMENİZ GEREKMEKTEDİR. KİSMİ PUANLAMA YAPILMAMAKTADIR. Örneğin, cevap 14.3 ise 14.30 yazan öğrenci o sorudan puan alamaz! / ve \ farkı, ' ve " farkı, $\leq$ yerine $\leq$ gibi detaylar da dâhil olmak üzere CEVABINIZIN TAM OLARAK DOĞRULUĞUNA emin olunuz. Ancak .(nokta) yerine ,(virgül) koymak ya da C yerine c yazmak gibi durumlar, soruda özellikle ölçülenler arasında değilse, puan kaybına yol açmaz.
- EKRAN ÇIKTISI SORULARINDA KODLARIN ÇIKTISINI HER HÜCRE BİR KARAKTERİ TEMSİL EDECEK ŞEKİLDE KUTULARA YAZINIZ. Cevabınızın kutuya sığmadığı durumlar olursa, ilgili kısma ek hücreler çizerek cevabınızı belirtebilirsiniz. Hata içerdiğini düşündüğünüz sorularda (mantık hatası haricindekilerde) ne tür bir hata olduğunu aşağıdaki seçeneklerden seçerek cevap kutusunun içine (örn. "HATA1" şeklinde) yazınız VE nedenini cevap kutusunun altına ya da soruya yakın başka boş bir alana (örn. "Çünkü koddaki ... eksik." şeklinde) belirtiniz. HATA1:Kod hiç derlenemeyecektir. HATA2:Kod derlenecek ama çalışması esnasında hata verecektir. HATA3:Kod derlenecek ama çalışması esnasında sonsuz döngüye girecektir. Yukarıdaki ifadelere uymayan hatalarda (örneğin mantık hatası içerdiğini düşündüğünüz sorularda) hatasız sorularda olduğu gibi sadece ekran çıktısını yazınız, HATA# vs. yazmayınız.
- KODDA BOŞLUK SORULARINDA EKRAN ÇIKTISINI SAĞLAYACAK ŞEKİLDE BOŞLUĞU KULLANDIĞINIZ TÜM KARAKTERLERE ($\leq$ yerine $\leq$ gibi) DİKKAT EDEREK DOLDURUNUZ. Gereksiz komut/ifadeler eklemeyiniz; bu tür durumlarda kodunuz çıktı açısından isteneni yapıyor olsa bile puan alamayabilirsiniz.

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

```
typedef struct
```

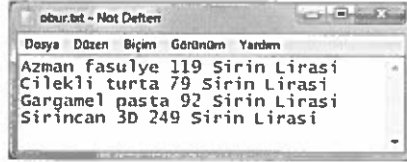
```
{  
    char isim[10][10][100];  
}
```

```
} data;
```

```
int main()
```

```
{  
    data veri;  
    FILE* dosya = fopen("obur.txt", "r");  
    int a = 8, k = 5;  
    while(a>=1)  
    {  
        fscanf(dosya,"%s", veri.isim[a-2][k%3]);  
        k++;  
        --a;  
    }  
    printf("%s\n", veri.isim[6][2]);  
    printf("%s\n", veri.isim[4][1]);  
    return 0;  
}
```

A	z	m	a	n					
1	1	9							



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

```
typedef struct
```

```
{  
    int a;  
} y;  
int f(y x);  
int main()  
{  
    y z;  
    z.a=7;  
    f(z);  
    printf("%d", z.a);  
    return 0;  
}  
int f(y x)  
{  
    x.a = x.a/3;  
    return x.a;  
}
```

7									

KODDA BOŞLUK DOLDURMA SORUSU

/ Çifte harfli, aynı harfi(küçük ya da büyük harf şeklinde) peş peşe içermenin en az 1 kez gerçekleştiği sözcüklere (örn. saat, cool, allık) deniliyor olsun. */*

```
#include <stdio.h>
#include <string.h>
int cifteHarf(char *ptr);
void kucult(char[]);
int main()
{
    char giris[100];
    printf("Sozcuk:");
    scanf("%s", giris);
    kucult(giris);
    if(cifteHarf(giris))
        printf("Cifte harfli");
    else
        printf("Cifte harfsiz");
    return 0;
}
int cifteHarf(char *ptr)
{
    int c, l = (int)strlen(ptr);
    for(c=0; c<l-1; c++)
    {
        if( ( *(ptr+c) ) == ( *(ptr + c + 1) ) )
            return 1;
    }
    return 0;
}
void kucult( char input [ ] )
{
    // 'A'nın ASCII'si 65, 'a'nın ASCII'si 97 olmaktadır.
    int i;
    for(i=0; input[i]!='\0'; i++)
        input[i] = (input[i]<'a')?(input[i]+'a'-'A'):input[i];
}
```

ÖRNEK EKRAN ÇIKTISI

Sozcuk:kubBe Cifte harfli ----- Process exited after	Sozcuk:boolean Cifte harfli ----- Process exited after
---	---

Soldaki çıktıda, kod çalıştırıldıktan sonra kullanıcı "kubBe" yazıp enter'a basmıştır. b harfi peş peşe geçtiği için, ikinci b harfi büyük olarak geçmesine rağmen istenen şekilde çifte harfli olarak algılanabilmiştir ve ekrana "Cifte harfli" yazdırılmıştır. Sağdaki çıktıda, kod çalıştırıldıktan sonra kullanıcı "boolean" yazıp enter'a basmıştır.

ÖRNEK EKRAN ÇIKTISI

Sozcuk:mustenkif Cifte harfsiz ----- Process exited afte

Kod çalıştırıldıktan sonra kullanıcı "mustenkif" yazıp enter'a basmıştır.

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

```
enum radyo {FM, AM, LW, MW, SW};
```

```
int main()
```

```
{
```

```
    printf("%d", AM);
```

```
    //printf("%d", AM);
```

```
    printf("%d", SW-FM);
```

```
    return 0;
```

```
}
```

1	4								

BU KISA SINAVDA 2 SORUDAKİ KODLAR DEVCPP'TAKİ GİBİ TONLAMALI OLARAK BASTIRILMIŞTIR. DÖNEM SONU SINAVINDA DA KODLAR BU ŞEKİLDE BASTIRILABİLİR. TERCİHİNİZİ İŞARETLEYİNİZ VE DİLERSENİZ YORUMLARINIZI EKLEYİNİZ:

a) (Bundan önce olduğu gibi) bütün karakterlerin aynı tonda olmasını tercih ederim.

b) Tonlamalı (bazı karakterler **koyu**, bazıları *gri*...) olmasını tercih ederim:

c) Benim için ikisi de olur. Fark etmez.

Yorumunuz:

ÖĞRENCİ İSMİ:

SÜRENİZ 10 DAKİKADIR.

DERS NO(3 basamaklı):

BİL141 KISA SINAV V ŞUBE 2 GRUP M SORULARI

- ASİSTANLARIN SINAV ESNASINDAKİ TEK GÖREVİ SINIFI GÖZETLEMektir. SORULARDAN PUAN ALABİLMENİZ İÇİN TAM OLARAK CEVAP VERMENİZ GEREKMEKTEDİR. KİSMİ PUANLAMA YAPILMAMAKTADIR. Örneğin, cevap 14.3 ise 14.30 yazan öğrenci o sorudan puan alamaz! / ve \ farkı, ' ve " farkı, $\leq$ yerine $\leq$ gibi detaylar da dâhil olmak üzere CEVABINIZIN TAM OLARAK DOĞRULUĞUNA emin olunuz. Ancak .(nokta) yerine ,(virgül) koymak ya da C yerine c yazmak gibi durumlar, soruda özellikle ölçülenler arasında değilse, puan kaybına yol açmaz.
- EKRAN ÇIKTISI SORULARINDA KODLARIN ÇIKTISINI HER HÜCRE BİR KARAKTERİ TEMSİL EDECEK ŞEKİLDE KUTULARA YAZINIZ. Cevabınızın kutuya sığmadığı durumlar olursa, ilgili kısma ek hücreler çizerek cevabınızı belirtebilirsiniz. Hata içerdiğini düşündüğünüz sorularda (mantık hatası haricindekilerde) ne tür bir hata olduğunu aşağıdaki seçeneklerden seçerek cevap kutusunun içine (örn. "HATA1" şeklinde) yazınız VE nedenini cevap kutusunun altına ya da soruya yakın başka boş bir alana (örn. "Çünkü koddaki ... eksik." şeklinde) belirtiniz. HATA1:Kod hiç derlenemeyecektir. HATA2:Kod derlenecek ama çalışması esnasında hata verecektir. HATA3:Kod derlenecek ama çalışması esnasında sonsuz döngüye girecektir. Yukarıdaki ifadelere uymayan hatalarda (örneğin mantık hatası içerdiğini düşündüğünüz sorularda) hatasız sorularda olduğu gibi sadece ekran çıktısını yazınız, HATA# vs. yazmayınız.
- KODDA BOŞLUK SORULARINDA EKRAN ÇIKTISINI SAĞLAYACAK ŞEKİLDE BOŞLUĞU KULLANDIĞINIZ TÜM KARAKTERLERE ($\leq$ yerine $\leq$ gibi) DİKKAT EDEREK DOLDURUNUZ. Gereksiz komut/ifadeler eklemeyiniz; bu tür durumlarda kodunuz çıktı açısından isteneni yapıyor olsa bile puan alamayabilirsiniz.

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

```
enum radyo {FM, AM, LW, MW, SW};
```

```
int main()
```

```
{
```

```
printf("%d", LW);
```

```
//printf("%d", AM);
```

```
printf("%d", SW-AM);
```

```
return 0;
```

```
}
```

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

```
typedef struct
```

```
{
```

```
char isim[10][10][100];
```

```
} data;
```

```
int main()
```

```
{
```

```
data veri;
```

```
FILE* dosya = fopen("obur.txt", "r");
```

```
int a = 8, k = 5;
```

```
while(a>=1)
```

```
{
```

```
fscanf(dosya,"%s", veri.isim[a-2][k%3]);
```

```
k++;
```

```
--a;
```

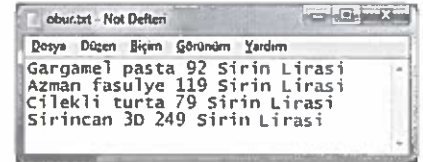
```
}
```

```
printf("%s\n", veri.isim[6][2]);
```

```
printf("%s\n", veri.isim[4][1]);
```

```
return 0;
```

```
}
```



2	3								

G	a	r	g	a	m	e	l		
9	2								

KODDA BOŞLUK DOLDURMA SORUSU

/* Çifte harfli, aynı harfi(küçük ya da büyük harf şeklinde) peş peşe içermenin en az 1 kez gerçekleştiği sözcüklere (örn. saat, cool, allık) deniliyor olsun. */

```
#include <stdio.h>
#include <string.h>
int cifteHarf(char *ptr);
void kucult(char[]);
int main()
{
    char giris[100];
    printf("Sozcuk:");
    scanf("%s", giris);
    kucult(giris);
    if(cifteHarf(giris))
        printf("Cifte harfli");
    else
        printf("Cifte harfsiz");
    return 0;
}
int cifteHarf(char *ptr)
{
    int k, l = (int)strlen(ptr);
    for(k=0; k<l-1; k++)
    {
        if( ( *(ptr+k) ) == ( *(ptr + k + 1) ) )
            return 1;
    }
    return 0;
}
void kucult( char input [ ] )
{
    // 'A'nın ASCII'si 65, 'a'nın ASCII'si 97 olmaktadır.
    int i;
    for(i=0; input[i]!='\0'; i++)
        input[i] = (input[i]<'a')?(input[i]+'a'-'A'):input[i];
}
```

ÖRNEK EKRAN ÇIKTISI

Sozcuk:kubBe Cifte harfli ----- Process exited after	Sozcuk:boolean Cifte harfli ----- Process exited after
---	---

Soldaki çıktıda, kod çalıştırıldıktan sonra kullanıcı "kubBe" yazıp enter'a basmıştır. b harfi peş peşe geçtiği için, ikinci b harfi büyük olarak geçmesine rağmen istenen şekilde çifte harfli olarak algılanabilmiştir ve ekrana "Cifte harfli" yazdırılmıştır. Sağdaki çıktıda, kod çalıştırıldıktan sonra kullanıcı "boolean" yazıp enter'a basmıştır.

ÖRNEK EKRAN ÇIKTISI

Sozcuk:mustenkif Cifte harfsiz ----- Process exited after
--

Kod çalıştırıldıktan sonra kullanıcı "mustenkif" yazıp enter'a basmıştır.

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

```
typedef struct
```

```
{
    int a;
} y;
int f(y x);
int main()
{
    y z;
    z.a=9;
    f(z);
    printf("%d", z.a);
    return 0;
}
int f(y x)
{
    x.a = x.a/4;
    return x.a;
}
```

9									

BU KISA SINAVDA 2 SORUDAKİ KODLAR DEVCPP'TAKİ GİBİ TONLAMALI OLARAK BASTIRILMIŞTIR. DÖNEM SONU SINAVINDA DA KODLAR BU ŞEKİLDE BASTIRILABİLİR. TERCİHİNİZİ İŞARETLEYİNİZ VE DİLERSENİZ YORUMLARINIZI EKLEYİNİZ:

a) (Bundan önce olduğu gibi) bütün karakterlerin aynı tonda olmasını tercih ederim.

b) Tonlamalı (bazı karakterler **koyu**, bazıları *gri*...) olmasını tercih ederim:

c) Benim için ikisi de olur. Fark etmez.

Yorumunuz: