

BM102 - ALGORİTMA VE PROGRAMLAMA – 2

ÖDEV -1

AÇIKLAMALAR;

- Ödevler Bireysel olarak yapılacaktır.
- https://turnitin.com/login_page.asp?lang=tr sitemine yüklenecektir.
- Turnitin kullanımı <http://akademik.duzce.edu.tr/arafatsenturk/Profil/Dokumanlar> adresinde mevcuttur.
- Sınıf Numarası: **25888105** Sınıf Şifresi:**2468**'dir.
- Ödevler her hangi bir derleyici (DevC gibi) kullanılarak yapılacaktır
- PDF dosyasının ismi şu şekilde olacaktır: ÖğrenciNO.pdf
Bu standartta gönderilmeyen ödevler değerlendirilmeye alınmayacaktır.
- Turnitin'a yükleme yaparken BAŞLIK bölümü de öğrenci No giriniz.
- Turnitin sistemindeki Ödev 1'in son yüklenme tarihi **21 Ağustos 2020 saat: 08:00'dır.** Bu tarih ve saatten sonra sistem otomatik olarak kapılacaktır.
- Turnitin sisteminin dışında e-mail veya elden teslim edilen ödevler kesinlikle kabul **EDİLMEYECEKTİR.**

ÖDEV -1

1. Aşağıda verilen programda 5 elemanlık bir tamsayı dizisi yaratılmıştır. Programın çalıştırılması sonucunda bellek tablosu üzerinde dizi elemanlarının değerlerini belirtiniz.

```
#include <iostream>
using namespace std;
int main ()
{
    int dizi[5];
    int *gosterge;
    gosterge = dizi;
    *gosterge = 5;
    gosterge++;
    *gosterge = 15;
    gosterge = &dizi[2];
    *gosterge = 20;
    gosterge = dizi + 3;
    *gosterge = 25;
    gosterge = dizi;
    *(gosterge+4) = 30;
    for (int i=0; i<5; i++)
        cout <<dizi[i]<<" ";
    cout<<endl;
    return 0;
}
```

dizi →

	Adres	Deger
[0]	0x22ff50	
[1]	0x22ff54	
[2]	0x22ff58	
[3]	0x22ff62	
[4]	0x22ff66	

2. Aşağıdaki programın çalışma zamanında değişkenlerin bellekteki yerleşimi verilmiştir.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 29;
    int b = 25;
    int *x;
    int *y;
    ...
}
```

	Adres	Deger
...	...	
y	0x22ff68	
x	0x22ff6c	
b	0x22ff70	29
a	0x22ff74	25

Programın devamında aşağıda verilen komutlar çalıştırıldığında bellekteki değerlerin son durumunu ve programın çıktısını yazınız.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 29;
    int b = 25;
    int *x;
    int *y;
    x = &a;
    y = &b;
    *x = *y + 3;
    *y = *x + 2;
    cout<<"x:"<<x<<endl;
    cout<<"*x:"<<*x<<endl;
    cout<<"&x:"<<&x<<endl;
    cout<<"y:"<<y<<endl;
    cout<<"*y:"<<*y<<endl;
    cout<<"&y:"<<&y<<endl;
    return 0;
}
```

3. Kullanıcının belirleyeceği sayı adedi kadar sayıyı okuyan ve en büyük sayıyı ekranda gösteren bir program yazılmış ancak programın bazı kısımları eksik bırakılmıştır. Aşağıda gösterilen örnek çıktıyı dikkate alacak şekilde, eksikleri tamamlayınız.

```
#include <iostream>
using namespace std;
void sayiOku(_____, (_____));
void yazdir(_____, (_____));
float maksimumHesapla((_____, (_____));
int main ()
{
    (_____ g;
    int n;
    float ortalama, max, min;
    cout<<"Girmek istediginiz sayi adedini giriniz:";
    cin>>n;
    g = (_____ (_____ [n];
    sayiOku(g, n);
    yazdir(g, n);
    cout<<"En buyuk sayi:"<<maksimumHesapla(g,n)<<endl;
    delete[](_____);
    return 0;
}

void sayiOku((_____ g, int n)
{
    for(int i=0; i<n; i++){
        cout<<i+1<<" . sayi:";
        cin>>(_____ ;
    }
}
void yazdir((_____ g, int n)
{
    for(int i=0; i<n; i++){
        cout<<(_____ <<endl;
    }
}
_____ maksimumHesapla((_____ g, int n)
{
    (_____ m = *g;
    for(int i=0; i<n; i++){
        if((_____ >m){
            m = (_____ ;
        }
    }
    _____;
}
}
```

Örnek Çıktı

```
Girmek istediginiz sayi adedini giriniz: 5
1.sayi: 89.45
2.sayi: 34.56
3.sayi: 67.98
4.sayi: 95.7
5.sayi: 89.57
89.45
34.56
67.98
95.7
89.57
En buyuk sayi: 89.57
```

4. Aşağıda tanımı ve ilk değer ataması yapılan x, y ve z değişkenleri verilmiştir. Bu değişkenlere ait bellekteki değerlerin aşağıdaki kurallara göre değiş tokuş yapacak `degistir()` fonksiyonunu göstergeler kullanarak yazınız.

```
x->z
z->y
y->x
int main ()
{
    int x = 10;
    int y = 30;
    int z = 40;
    degistir(&x, &y, &z);
    cout<<x<<endl;
    cout<<y<<endl;
    cout<<z<<endl;
    return 0;
}
```

Örnek Çıktı

```
30
40
10
```

5. Yukarıdaki soruda yazdığınız fonksiyonu, parametre olarak gönderilen dizinin ilk ve son elemanları hariç kopyalayan bir fonksiyon olarak yeniden yazınız.

Örnek Çıktı

```
Dizinin boyutunu giriniz: 5
Dizinin elemanlarını giriniz: 2 4 6 8 10
Dizinin ilk ve son elemansız kopyası: 4 6 8
```

6. Diyetisyenler, kişinin ihtiyacına ve metabolizma özelliklerine göre kişiye uygun farklı diyet programları uygularlar. Örnek çıktıda da görüldüğü gibi, kişinin takip ettiği diyet programı sonucunda ne kadar kilo kaybettiğini gösteren bir program yazınız. Bu programda dinamik bellek kullanmanız istenmektedir.

Örnek Çıktı

```
Diyet programınızın süresini giriniz (hafta sayısı): 5
1.hafta kilonuz: 72.3
2.hafta kilonuz: 71.1
3.hafta kilonuz: 69.5
4.hafta kilonuz: 67.8
5.hafta kilonuz: 66.1
Bu diyet programı ile toplam 6.2 kilo kaybettiniz.
```

7. İstatistikte **mod** bir veri serisi içinde en çok tekrar eden sayıdır. Örneğin aşağıdaki seriyi ele alalım.

3,5,6,5,7,8,7,5,8,5,9

Bu seride en çok tekrar eden sayı 5 (4 kez) olduğundan bu serinin modu 5'tir. Eğer bir serinin modu yoksa, bir başka deyişle tüm değerler sadece 1 kez tekrarlıyorsa, serinin modunu -1 olarak alabiliriz. Her seride tek bir mod olabileceğini ve tüm değerlerin tek basamaklı olacağını varsayarak, bir dizinin modunu bulan bir program yazınız. Programınız aşağıdaki bileşenlerden oluşsun:

- `mod()`: bir diziyi ve boyutunu parametre olarak alıp modunu döndürür.
- `main()`: kullanıcıdan veri serisi boyutunu okuyup, dinamik bir diziye tek basamaklı tamsayı değerlerini okur, daha sonra `mod()` fonksiyonunu çağırıp serinin modunu ekrana yazdırır.

Örnek Çıktı

```
Veri serisinin boyutunu giriniz: 10
Veri serisini giriniz: 1 3 2 1 4 2 6 3 1 8
Mod: 1
```

8. A sınıfında tanımlı değişken ve fonksiyonların sınıf dışından erişilebilirliklerini gösteren tablo aşağıda verilmiştir. Bu tabloya uygun olarak A sınıfını değişken ve fonksiyonları ile tanımlayınız.

	x	Y	f1()	f2()
A sınıfı içinden erişim	✓	✓	✓	✓
A sınıfı dışından erişim		✓		✓

9. Aşağıda verilen programın çıktısını yazınız.

```
#include <iostream>
using namespace std;
class A
{
    int f1();
public:
    int x;
    int y;
    int f2();
    void xAta(int x);
};

int A:: f1()
{
    return 2*x*y;
}
int A:: f2()
{
    f1();
    return x*x+y*y;
}
void A:: xAta(int _x)
{
    x = _x;
}
int main()
{
    A a[3];
    int x,y;
    cout<<"x*x+2*x*y+y*y = ?"<<endl;
    for(int i=0; i<3; i++){
        a[i].x = i*2+1;
        a[i].y = i+2;
    }
    for(int i=0; i<3; i++){
        x = a[i].x;
        y = a[i].y;
        cout<<x<<"*"<<x<<"+"2*"<<x<<"*"<<y<<"+"
            <<y<<"*"<<y<<"="<<a[i].f2()<<endl;
    }
    return 0;
}
```


10. Aşağıda verilen programın çıktısını yazınız.

```
#include <iostream>
using namespace std;
int a=3;
class Test
{
    int b,c;
public:
    void atama(int,int);
    int fon(int&, int=8);
};

void Test::atama(int d,int e)
{
    b = d;
    c = e;
    a += 5;
}
int Test::fon(int& f, int g)
{
    f*=3;
    g*=4;
    b+=2;
    c+=3;
    cout<<f<<" "<<g<<endl;
    return f+g;
}
int main()
{
    int h=5, i=6;
    Test nsn1,nsn2;
    nsn1.atama(1,3);
    nsn2.atama(2,4);
    a=nsn1.fon(h,i);
    cout<<h<<" "<<i<<" "<<a<<endl;
    i=nsn2.fon(h);
    cout<<h<<" "<<i<<" "<<a<<endl;
    return 0;
}
```

11. Aşağıda bir saatin çalışmasını gösteren bir program verilmiş ancak bazı kısımlar eksik bırakılmıştır. Bu programda Saat isimli sınıfın iki veri üyesi olarak saat ve dakika tanımlanmıştır. Saat sınıfın üye fonksiyonları aşağıda verilmiştir:

- kur(): gönderilen parametreleri saat ve dakika üyelerine atar. Parametre yollanmamışsa varsayılan değerleri (saat: 12, dakika: 0) kullanır.
- arttır(): şu andaki saati bir dakika arttırır.
- goster(): şu andaki saati ss:dd formatında yazdırır (örneğin 14:30)

main() fonksiyonu Saat tipinde bir nesne yaratıp, 14:55'e kurar ve sonraki 10 dakika boyunca saati yazar. Programın örnek çıktıyı verecek şekilde çalışması için eksikleri tamamlayınız.

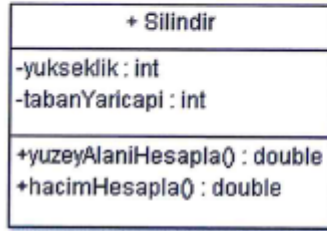
Örnek Çıktı

```
Saat: 14:55
10 dakika boyunca:
14:55
14:56
14:57
14:58
14:59
15:00
15:01
15:02
15:03
15:04
```

```
#include <iostream>
using namespace std;
class Saat
{
    int saat, dakika;
public:
    void basla(_____, _____);
    _____;
    void goster();
};
void _____::basla(_____, _____)
{
    saat=s;
    dakika=d;
}
void Saat::arttir()
{
    if (dakika<59)
        _____;
    else
    {
        _____;
        if (saat<23)
            saat++;
        else
            _____;
    }
}
void _____
{
    if (saat==0)
        cout<<"00:";
    else
        cout<<"_____";
    if (dakika==0)
        cout<<"00"<<endl;
    else if (_____)
        cout<<"0"<<dakika<<endl;
    else
        cout<<dakika<<endl;
}
int main()
{
    Saat s;
    s.basla(_____, _____);
    cout<<"Saat:";
    _____;
    cout<<"10 dakika boyunca:"<<endl;
    for (int i=1;i<=10;i++){
        _____;
        _____;
    }
    return 0;
}
```


12. **DERSTE ÇÖZÜMÜ YAPILDI, CEVAPLARI WEBDE VERİLDİ.**

Silindir sınıfına ait sınıf diyagramı ve bu sınıfa ait fonksiyonların gerçekleştirilmesinde kullanılacak formüller aşağıda verilmiştir. Program çalıştırıldığında aşağıdaki örnek çıktı elde edilecek şekilde Silindir sınıfını yazınız.



$$A = 2\pi rh + 2\pi r^2$$

$$H = \pi r^2 h$$

Örnek Çıktı

```
Yukseklik (h):5
Taban yarıcapı (r) :3
-----
Yuzey alani (A):150.796
Hacmi (H):141.372
```

13. Bir yerleşim biriminde yıllık doğum oranı ve ölüm oranı aşağıdaki formüllerle hesaplanır:

$$\text{Doğum Oranı} = \text{Doğum Sayısı} / \text{Nüfus}$$

$$\text{Ölüm Oranı} = \text{Ölen Sayısı} / \text{Nüfus}$$

Aşağıdaki bileşenlerden oluşan YerlesimBirimi isimli bir sınıf tasarlayın:

- Yerleşim biriminin ismi
- Yerleşim biriminin nüfusu
- Doğum ve ölen sayıları
- Kullanıcıdan verileri okuyan isimAta(), nufusAta(), dogumAta(), olumAta() fonksiyonları
- Doğum Oranını bulan bir fonksiyon
- Ölüm Oranını bulan bir fonksiyon

Programınız örnek çıktıda görüldüğü gibi Ankara ve Türkiye için doğum ve ölüm oranlarını hesaplayacaktır.

Örnek Çıktı

```
Yerlesim Biriminin ismini giriniz: Ankara
Nufusunu giriniz:4890893
Dogum sayisini giriniz: 67660
Olum sayisini giriniz: 20634
Ankara --- Dogum Orani:0,0138339
Ankara --- Olum Orani: 0,00421886
Yerlesim Biriminin ismini giriniz: Turkiye
Nufusunu giriniz: 74724269
Dogum sayisini giriniz: 1238970
Olum sayisini giriniz: 365190
----- Dogum Orani:0,0165806
----- Olum Orani: 0,00488717
```

14. **DERSTE ÇÖZÜMÜ YAPILDI, CEVAPLARI WEBDE VERİLDİ.**

Bir kola makinesi için program yazalım. Programda KolaMakinesi isimli aşağıdaki bileşenlerden oluşan bir sınıf tanımlayın:

- Veri Üyeleri:
 - Bir kolanın fiyatı
 - Makinada kalan kolaların sayısı
- Üye fonksiyonlar:
 - ilkYukleme(): fiyat ve kola sayısına parametre olarak gelen değerleri atar. Varsayılan fiyat değeri 2.5 lira, sayı değeri ise 100 olarak alınır.
 - kolaVerme(): kullanıcıya fiyatı gösterir. Kullanıcı yeterli para girdiyse ve makinede kola varsa kullanıcıya bir kola verir ve kullanıcının para üstünü hesaplar.
 - goster (): Kola fiyatını ve makinede kalan kola sayısını yazar.

main() fonksiyonunda okulunuz için bir kola makinesi nesnesi yaratın, bir kullanıcının kola alması durumunda makinenizi örnek çıktıda olduğu gibi güncelleyin.

Örnek Çıktı

```
Kolanin fiyatı: 2.5 lira.
Odemeniz: 3
Kolaniz veriliyor.
Para ustunuz: 50 kurus
Son durum:
    Kolanin fiyatı: 2.5 lira.
    Kalan Kola sayisi : 99
```

15. Lirayı kuruşa (50, 25, 10, 5, 1 kuruş) çeviren bir program yazınız. Programınız aşağıdaki bileşenlerden oluşacaktır:

- Para isimli bir sınıf:
 - Veri üyeleri: liraDegeri, elli, yirmiBes, on, bes, bir kuruş değeri
 - Üye fonksiyonlar:
 - liraAta(): liraDegerine gelen parametreyi atar
 - donustur(): liraDegerini kuruşa çevirir
 - yaz(): paranın kuruş değerini yazdırır.
- main() fonksiyon: girilen paranın kuruş değerlerini örnek çıktıda olduğu gibi yazdırır.

Örnek Çıktı

```
Para miktarini lira olarak giriniz: 3.74
3.74 lira:
7 50 kurus,
0 25 kurus,
2 10 kurus,
0 5 kurus,
4 1 kurus
```

16.

Telsizler, alıcı-verici olmak üzere iki yönlü iletişim sağlayan taşınabilir cihazlardır. Telsizlerle konuşma ve dinleme iletişim fonksiyonları gerçekleştirilir. Örnek bir resmi de verilen basit bir telsizde aşağıdaki bileşenler yer alır:

- Ses seviyesini ayarlama ve aynı zamanda açma/kapama işlevini yapan ses ayar anahtarı (ses seviyesi 0 olduğunda kapama, 1 olduğunda açma işlemi yapar),
 - Frekans/kanal ayarlamayı sağlayan frekans ayar anahtarı,
 - Güç modunun ayarlandığı güç ayar anahtarı,
 - Telefonlarda olduğu gibi karşılıklı konuşma yerine sıra ile konuşmanın geçerli olmasını sağlayan konuşma mandalı
 - Konuşma mandalına basılı olduğunu kırmızı ile gösteren ışık göstergesi,
 - Aynı frekansta başkalarının görüşme yaptığını sarı ile gösteren ışık göstergesi,
- a. Telsiz sınıfını UML sınıf diyagramını çizerek modelleyiniz.
- b. Telsiz sınıfını, aşağıda main() fonksiyonu verilen programın örnek çıktıdaki gibi çalışmasını sağlayacak şekilde programlayınız.



```
int main()
{
    Telsiz telsiz;
    telsiz.ac();
    telsiz.sesAyarla(10);
    telsiz.kanalAyarla(32.4);
    telsiz.konus("Trafik kazasi var, yardim gonderin...");
    telsiz.ledGoster();
    string mesaj = telsiz.dinle();
    cout<<">>>> Gelen mesaj:"<<mesaj<<endl;
    telsiz.konus("Tamam...");
    telsiz.kapa();
    cout<<"Ses seviyesi:"<<telsiz.ses<<endl;
    return 0;
}
```

Örnek Çıktı

```
Trafik kazasi var, yardim gonderin...
Kirmizi LED yaniyor...
112 arandi
Gelen mesaj: 112 arandi
Tamam...
Ses seviyesi:0
```