

Pascal Programlama Ders Notları

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

Programlama Temel Kavramlar

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

Temel Kavramlar

- **Programlama dili:** bir programcının bir bilgisayara ne yapmasını istediğini anlatmasının yoludur.
- **Kaynak kod:** herhangi bir yazılımın işlenip makine diline çevrilmeden önce programcının okuyup üzerinde çalışabildiği programlama diliyle yazılmış hali.
- **Derleyici:** bir programlama dilinde yazılmış olan kaynak kodu makine dilene çeviren yazılım.

Temel Kavramlar

- **Algoritma**: matematikte veya bilgisayarda, bir işi yapmak için başlangıcı ve sonu belli olan, sonlu işlemler/adımlar kümesi.
- **Akış şeması**: algoritmaların grafiksel olarak ifade edilmesi

Bilgisayar ile Problem Çözmek

- Problemin tanımlanması
- Çözüm yollarını belirleme ve en uygun çözümün seçilmesi
- Çözüm yolunun **Algoritmasının** belirlenmesi
- Akış şemasının çizilmesi
- Programın kodlarının yazılması
- Programın çalıştırılması varsa hataların düzeltilmesi
- Dökümantasyon

Örnek problem - 1

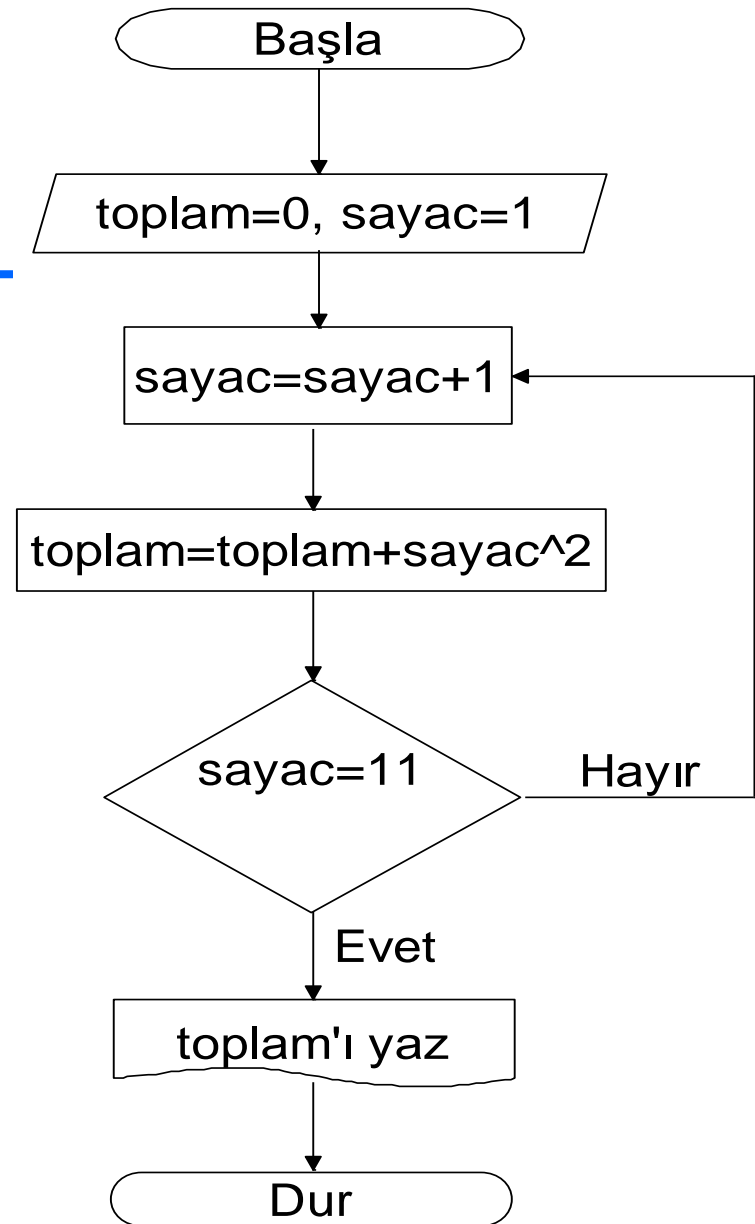
- 1'den 10'a kadar olan sayıların karelerinin toplamını hesaplayıp ekrana yazan bir programın algoritmasını yazınız, akış şemasını çiziniz.

Örnek problem - Algoritma:

1. Başla
2. toplam ve sayaç değişkenlerini tanımla
($\text{toplam}=0$, $\text{sayaç}=1$)
3. sayaç'ı bir arttır ($\text{sayaç}=\text{sayaç}+1$)
4. toplam'ın yeni değeri: kendi değeri ile
sayaç'ın değerinin karesinin toplamı
($\text{toplam}=\text{toplam}+\text{sayaç}^2$)
5. $\text{sayaç} \neq 11$ olduğu sürece 3. adıma git
6. toplam değerini ekrana yaz
7. Dur

Örnek problem

Akış şeması:



Pascal Programlama Dili

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

Pascal Programlarının Yapısı

Program program_adı;

Uses unitler

type Özel veri tipleri

Const Sabitler

Label etiketler

var değişken veya değişkenler

Tanımlama Bloğu

Begin

Ana Blok, Gövde

end.

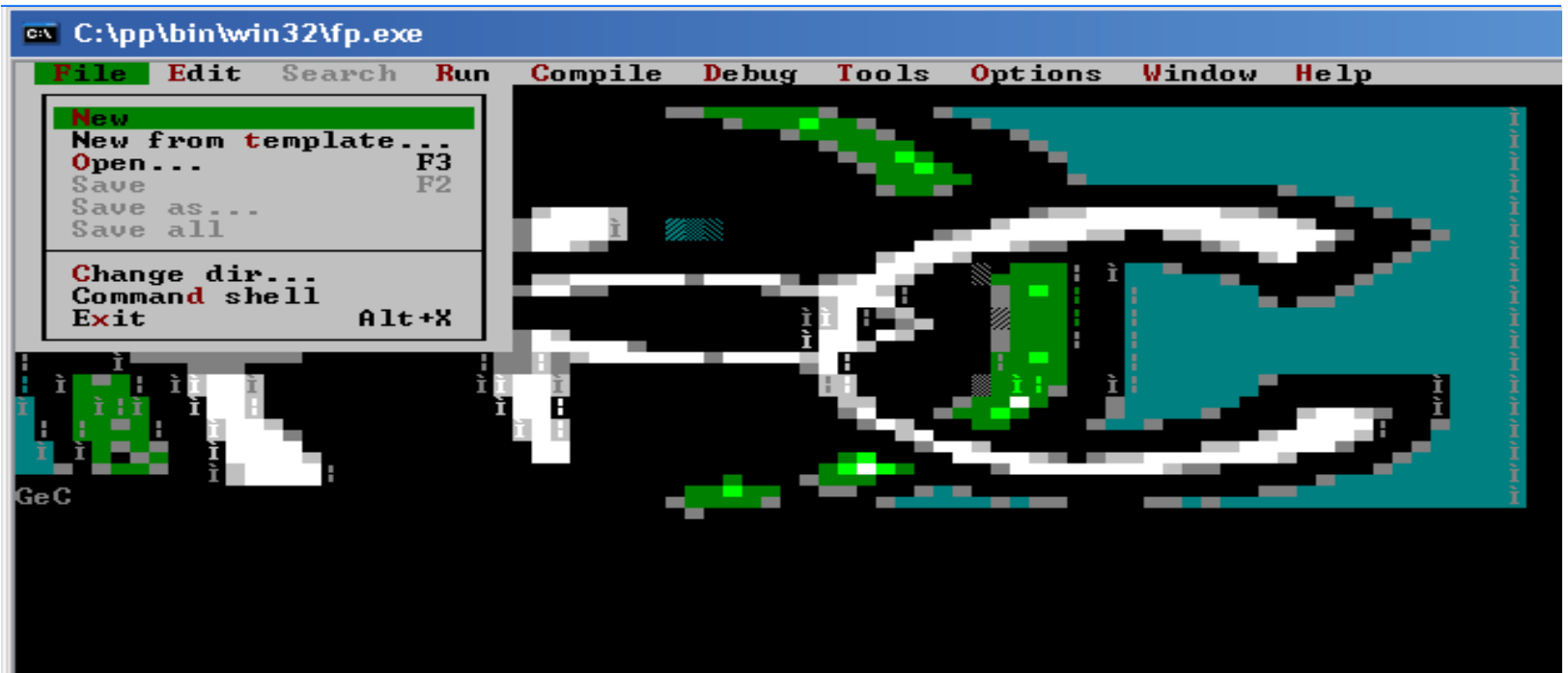
Pascal Programlarının Yapısı

- **Program komutu:** yazılan programa isim vermek için kullanılır.
Örnek: **Program** ilk_program
- **Uses komutu:** unit'leri (kütüphaneleri) kullanmak içindir. Örnek: crt (monitör ve klavyeyi kullanmak için)
- **Type komutu:** özel veri tipleri tanımlamak için
- **Const komutu:** sabit tanımlamak için. Örnek KDV, pi sayısı, e sayısı gibi
- **Label komutu:** etiket tanımlamak için.
- **Var komutu:** istenen tiplerde değişken tanımlamak için.

Free Pascal derleyicisi

- Derleyiciyi indirme adresi:
- <https://www.freepascal.org/download.html>
- Derleyici için kılavuz dosyası:
- <http://downloads.freepascal.org/fpc/docs-pdf/user.pdf>

Free Pascal derleyicisi



Yeni program dosyası oluşturma

İlk Programımız

Program ilk_program;

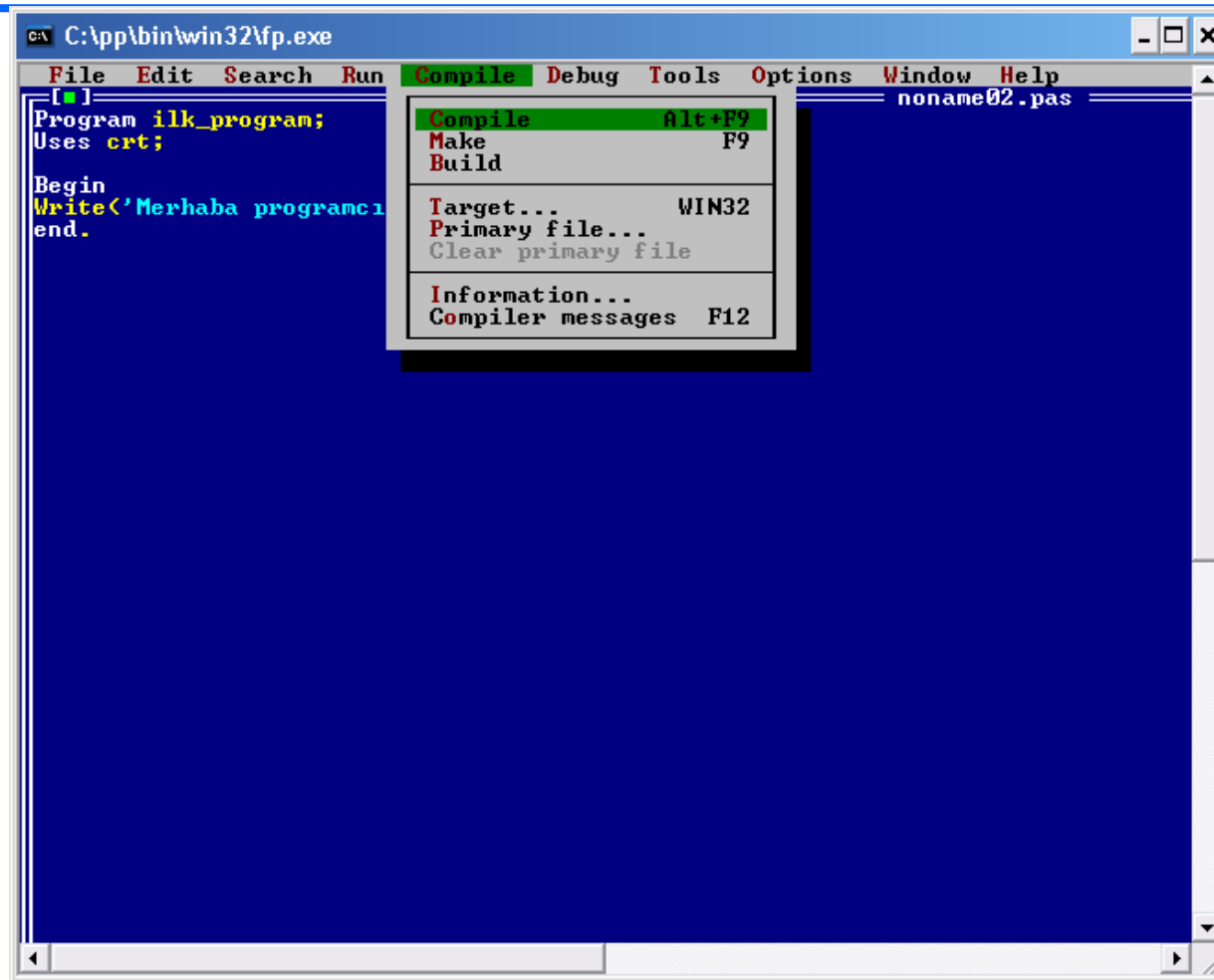
Uses crt;

Begin

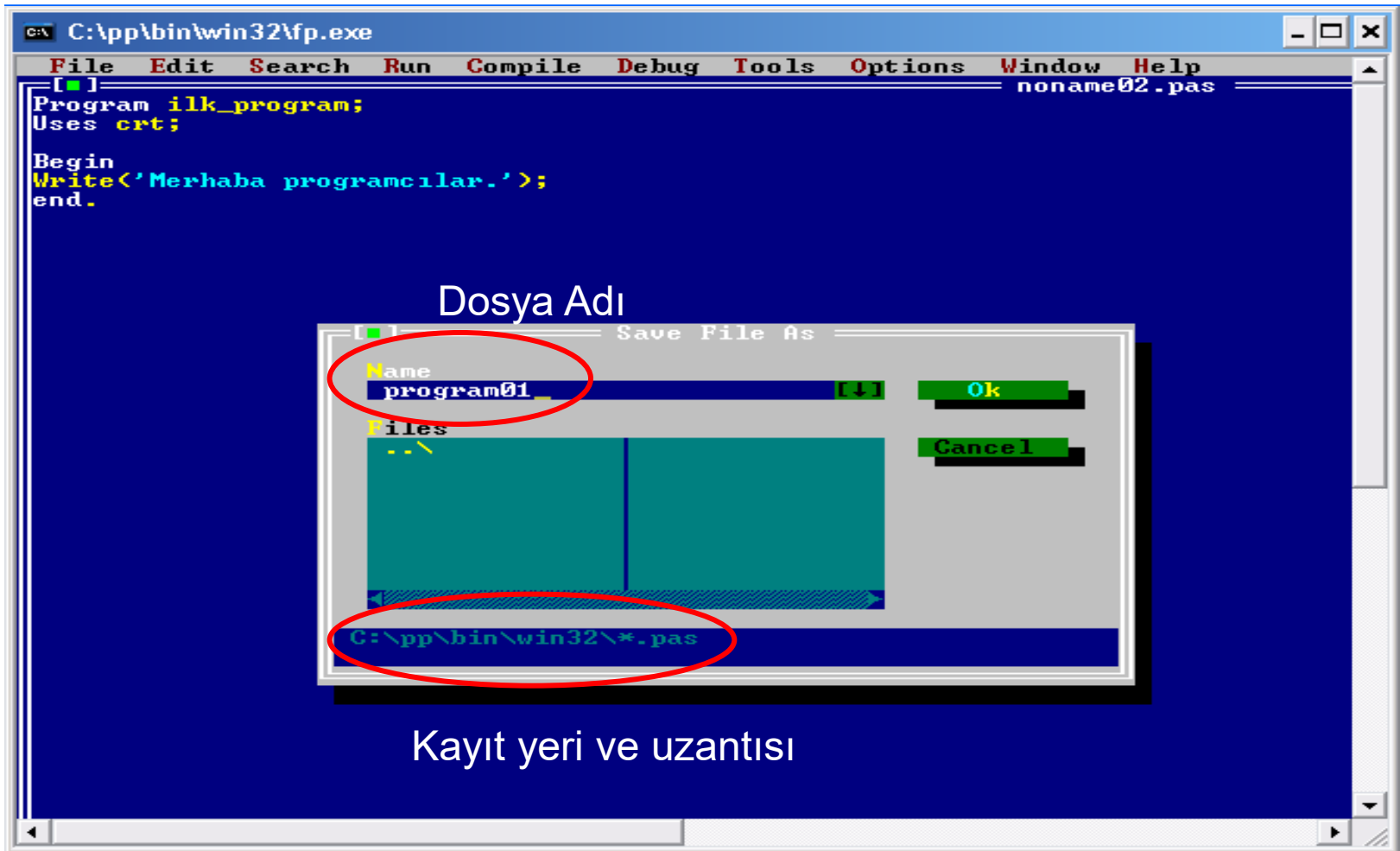
Write('Merhaba programcılar.');

end.

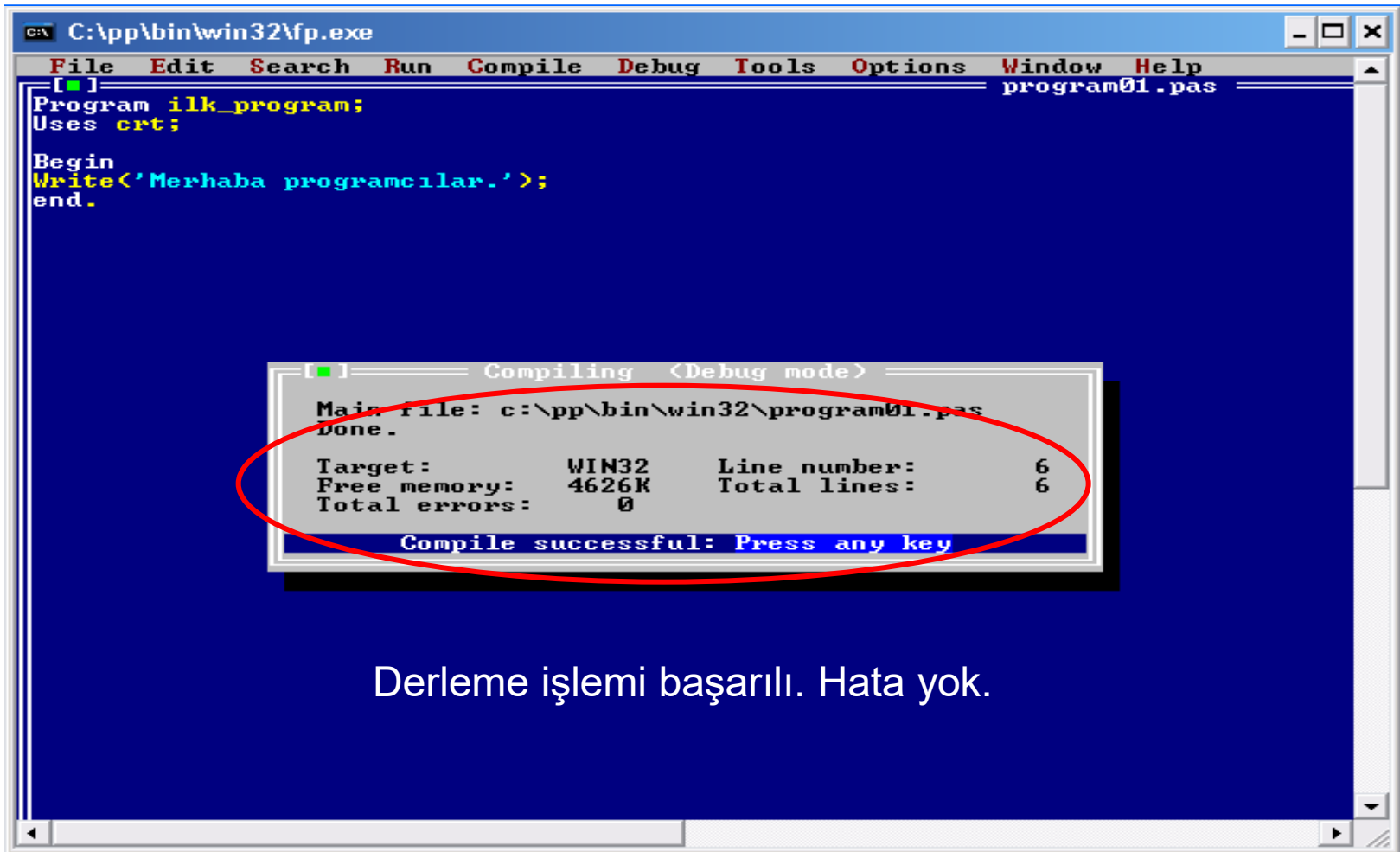
Programın Derlenmesi (Alt + F9)



Programın Kaydedilmesi



Derleme Sonuçları



```
File Edit Search Run Compile Debug Tools Options Window Help
[ ]
Program ilk_program;
Uses crt;

Begin
Write<'Merhaba programcılar.'>;
end.
```

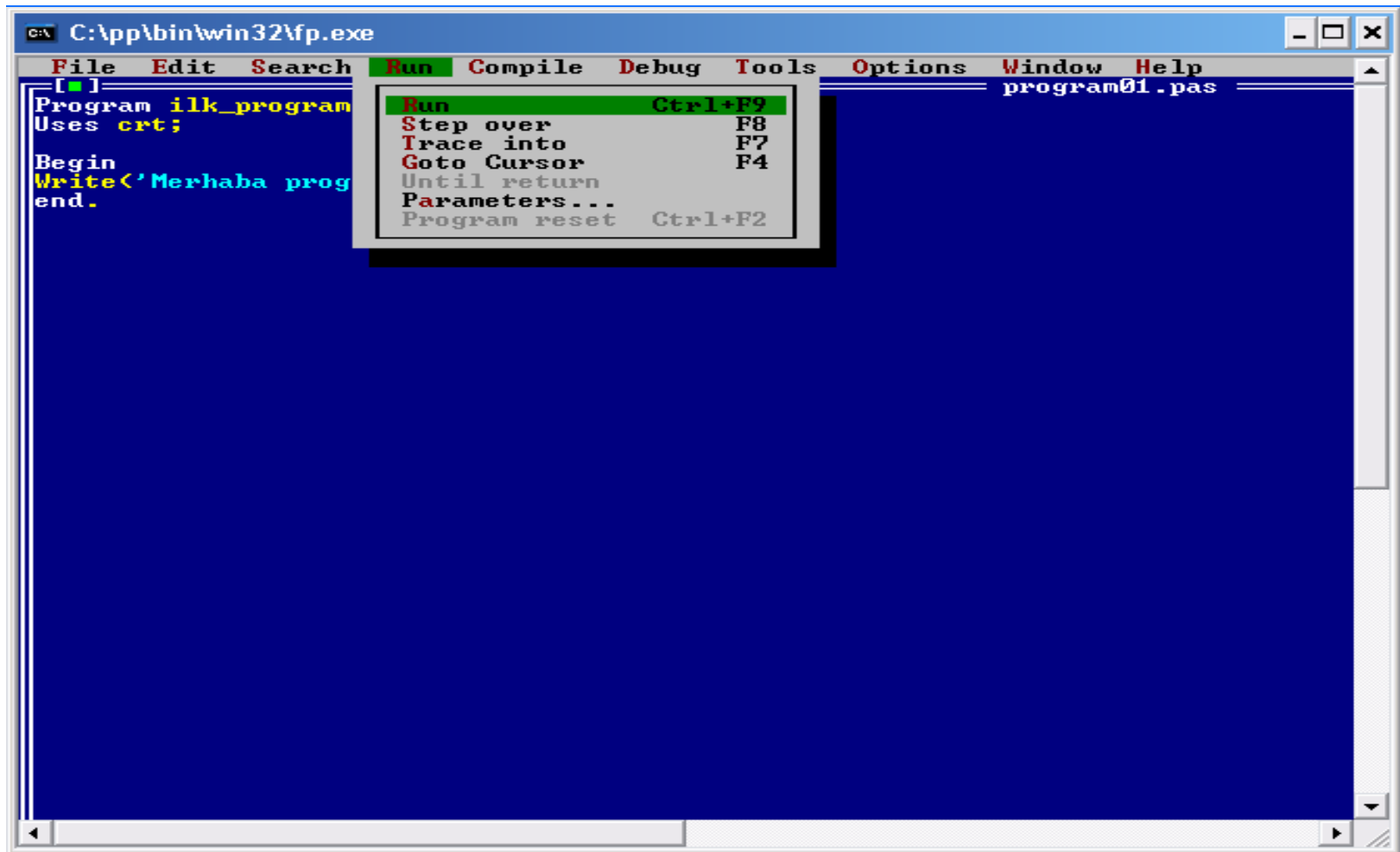
```
[ ] Compiling <Debug mode>
Main file: c:\pp\bin\win32\program01.pas
Done.

Target:      WIN32      Line number:      6
Free memory: 4626K      Total lines:      6
Total errors: 0

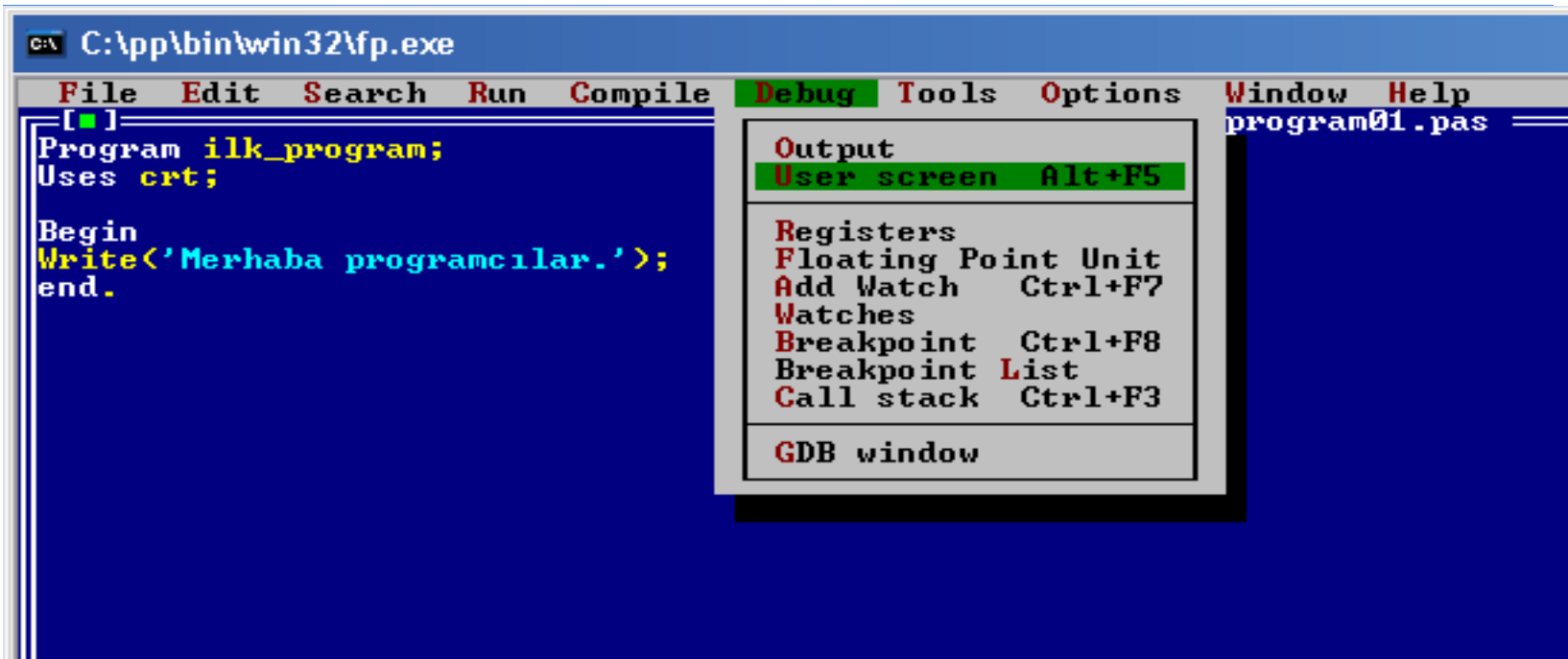
Compile successful: Press any key
```

Derleme işlemi başarılı. Hata yok.

Programın Çalıştırılması (CTRL + F9)



Kullanıcı Ekranı (Alt + F5)



Kullanıcı ekranı görüntüsü

C:\ C:\pp\bin\win32\fp.exe

```
■ Free Pascal IDE Version 0.9.2
Using "C:\pp\bin\win32\cygwin1.dll" version 1001.8.0.0
The cygwin1.dll that you have in "C:\pp\bin\win32\cygwin1.dll" is too old
If the IDE does not work correctly, please consider
putting a newer cygwin1.dll version in your path before that one.
Running "c:\pp\bin\win32\program01.exe "
Merhaba programcılar._
```

İlk Programın geliştirilmesi

Program ilk_program;

Uses crt;

Begin

Clrscr; {ekranı temizler}

Write('Merhaba programcılar.');

Readkey; {tuşa basılana kadar bekler}

end.

Giriş Çıkış Komutları

- **Write()**
- **Writeln()**

- **Read()**
- **Readln()**

TANIMLAMA BLOKLARI

- **Var**
- Kullanımı:
Değişken_adı:değişken_tipi;
- Örnek:
Var
x:integer;
ad:string[20];
ortalama:real;
sayi1,sayi2:word;

Tam Sayılar			
Tip	Alt Sınır	Üst Sınır	Bellekte Kapladığı Alan
ShortInt	-128	127	1 Byte
Integer	-32768	32767	2 Byte
Byte	0	255	1 Byte
Word	0	65535	2 Byte
LongInt	-2.147.483.648	2.147.483.647	4 Byte
Kesirli (Real) Sayılar			
Real	$2.9 * 10^{-39}$	$1.7 * 10^{38}$	6 Byte
Single	$1.5 * 10^{-45}$	$3.4 * 10^{38}$	4 Byte
Double	$5.0 * 10^{-324}$	$1.7 * 10^{308}$	8 Byte
Extended	$3.4 * 10^{-4932}$	$1.1 * 10^{4932}$	10 Byte
Comp	$-2^{63}+1$	$2^{63}-1$	8 Byte

Tip	Alt Sınır	Üst Sınır	Bellekte Kapladığı Alan
Char (Karakter)			
Char	#0	#255	1 Byte
Boolean (Mantıksal)			
Boolean	False	True	1 Byte
String (Karakter Dizisi)			
String	Karakter Dizisi		256 Byte

TANIMLAMA BLOKLARI

- **Const**
- Sabitleri tanımlamak için kullanılır.
Kullanımı:
Const
 Sabit_adı=değeri;
- Örnek:
Const
 KDV=0.18;
 e=2.71;

Kullanıcıya adını sorup onu selamlayan program

```
Program ikinci_program;  
Uses crt;  
Var ad:string[20];  
Begin  
  Clrscr;  
  Write('Lütfen adınızı yazınız: ');  
  Readln(ad);  
  Writeln('Merhaba ',ad);  
  Readkey;  
end.
```

Daire programı

- Kullanıcı tarafından yarıçapı verilen bir dairenin
 - Alanını
 - Çevresini
- Hesaplayıp ekrana yazdıran bir program yazınız...

Daire programı (Algoritması)

1. Yarıçap, alan ve çevre değişkenlerini tanımla.
2. Kullanıcıdan yarıçapı iste.
3. Yazılan değeri yarıçapa eşitle.
4. Alanı hesapla. ($\text{Alan} = \pi * r * r$)
5. Çevreyi hesapla. ($\text{Çevre} = 2 * \pi * r$)
6. Sonuçları ekrana yaz.

Dairenin alanını ve çevresini hesaplayan program

```
Program ikinci_program;  
Uses crt;  
Var r,alan,cevre:real;  
Begin  
  Clrscr;  
  Write('Lütfen dairenin yarıçapını yazınız: ');  
  Readln(r);  
  alan:=pi*r*r;  
  cevre:=2*pi*r;  
  Writeln('Daire alanı: ',alan:4:2,' , çevresi :',cevre:4:2);  
  Readkey;  
end.
```

Silindir programı (ÖDEV)

- Kullanıcı tarafından yarıçapı ve yüksekliği verilen bir silindirin
 - Alanını
 - Yanal alanını
 - Hacmini
- Hesaplayıp ekrana yazdıran bir programın algoritmasını ve pascal kodunu yazınız...

Karar verme yapıları

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

Karar verme yapıları - 1

- **If-Then-Else:** Belirtilen koşulun doğru olması durumunda belirtilen işlemin yapılmasını sağlamak için kullanılır.

Örnek:

IF ortalama \leq 60 **THEN** write('kaldı');

ELSE write('geçti');

Karar verme yapıları - 1

IF final<50 **THEN**

begin

writeln('finalden kaldı');

writeln('F3');

end

ELSE IF ortalama<=60 **THEN** writeln('F3')

ELSE IF ortalama<=65 **THEN** writeln('C')

ELSE IF ortalama<=70 **THEN** writeln('B2')

ELSE IF ortalama<=80 **THEN** writeln('B1')

ELSE IF ortalama<=90 **THEN** writeln('A2')

ELSE writeln('A1');

Not hesaplama programı (ÖDEV)

- Kullanıcıya vize ve final notunu soran
 - Ortalamayı
 - harfi
- hesaplayan programın algoritmasını ve pascal kodunu **if-then-else** kullanarak yazınız.
- $\text{Ortalama} = \text{vize} * 0.4 + \text{final} * 0.6$

2. Derece denklem (ÖDEV)

- $ax^2 + bx + c = 0$ denkleminin reel ve sıfırdan farklı olan katsayılarını kullanıcıdan isteyen
 - $a \neq 0$ olması şartıyla
 - Deltayı
 - X_1 ve x_2 'yi
- Hesaplayıp ekrana yazacak programın pascal kodunu yazınız.
- Karekök fonksiyonu: **sqrt()**
- $a=0$ olduğunda programın kilitlenmesini önlemek için **HALT** komutu ile programı sonlandırınız!!!

Karar verme yapıları - 2

- **Case - Of:** kullanım amacı if komutuyla hemen hemen aynıdır.

- Örnek

CASE ifade **OF**

Sabit1:komut veya komutlar

Sabit2:komut veya komutlar

:

Sabitn:komut veya komutlar

ELSE Komut veya komutlar

END

Karar verme yapıları - 2

CASE ay **OF**;

1:writeln('Ocak');

2:writeln('Şubat');

3:writeln('Mart');

:

12:writeln('Aralık');

ELSE writeln('Yanlış yazdınız.');

END

Ay programı (ÖDEV)

- Kullanıcı tarafından numarası girilen ayın
 - Adını
 - Mevsimini
- Bulan pascal programını iki ayrı **case-of** komutu kullanarak yazınız.

İşlem Sırası

Öncelik	İşlem
1	Not
2	*, /, Div, Mod, And
3	+, -, Or
4	<, <=, =, <>, >=, >

- Eğer aynı işlemciden birden fazla varsa işlem **soldan sağa** doğru yapılır.

Mod işlemi

```
Program mod_ornek;  
Uses crt;  
Var x : Integer;  
Begin  
  Clrscr;  
  Write('Bir sayı giriniz: ');  
  Readln(x);  
  If (x mod 2 = 1) Then Writeln ('Yazılan sayı tek  
    sayıdır.')  
  Else Writeln ('Yazılan sayı çift sayıdır.');
```

Readkey;
End.

Döngüler, Diziler

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

FOR DÖNGÜSÜ

- For döngüsü, bir program parçasının belirlenen sayıda üst üste çalıştırılması için kullanılır.

- **FOR** Başlangıç_Değeri **TO** Son_değer **DO**

Begin

İşlemler

End;

- **FOR** Başlangıç_Değeri **DOWNTO** Son_değer **DO**

Begin

İşlemler

End;

FOR Döngüsü ile Faktöriyel Programı

```
Program Faktoriyel;  
Uses crt;  
Var  
    sayac, sayi : Integer;  
    faktor : Extended;  
Begin  
    Clrscr;  
    Write('Faktöriyelini hesaplamak istediğiniz sayıyı yazınız: ');  
    Readln (sayi);  
    faktor := 1;  
    For sayac := 1 to sayi Do  
        faktor := faktor * sayac;  
    Writeln(sayi, ' Faktoriyel = ', faktor:10:0);  
    Readkey;  
End.
```

Repeat-Until Döngüsü

- Bir program bloğunun belli bir şart sağlanıncaya kadar üst üste çalıştırılmasını sağlar. Tekrar etme işi UNTIL deyimindeki boolean ifadesi sağlanıncaya kadar devam eder.

Repeat

```
_____  
_____  
_____
```

Until (Boolean Şartı)

Repeat-Until Döngü Örneği – Program tekrar çalışsın mı?

```
Program rep_until;  
Uses crt;  
Var  
    _____;  
    cevap : char;  
Begin  
Repeat  
    _____;  
    _____;  
    _____;  
    Write('Program tekrar çalışsın mı? (e/h) ');  
    Readln(cevap);  
Until cevap='h'  
End.
```

While-Do Döngüsü

- Bir program bloğunun belli bir şart sağlandığı sürece üst üste icrasını sağlar.

- **While** (Boolean Şartı) **Do**

Begin

```
_____;  
_____;  
_____;
```

End;

- WHILE ile REPEAT arasındaki fark: Repeat döngüsü şart cümlesi yanlış olduğu sürece, While döngüsü ise şart cümlesi doğru olduğu sürece çalışmaya devam eder.

While-Do döngüsü örneği

```
Program while_do;  
Uses crt;  
Var zaman:word;  
begin  
  Clrscr;  
  Writeln('Bu program 10 saniye sonra kendi kendini kapatacaktır');  
  zaman:=10;  
  while zaman>0 do  
    begin  
      write(zaman,' ');  
      Delay(1000);           {1000 milisaniye bekler.}  
      zaman:=zaman-1;  
    end;  
end.
```


Örnek Soru – Üçgen Programı

- Dik kenar uzunlukları verilen bir üçgenin hipotenüs uzunluğunu, çevresini ve alanını hesaplayan Pascal programını aşağıdaki uyarıları dikkate alarak yazınız:
- Program başlarken ekranı temizleyecek.
- Program kullanıcıdan a ve b pozitif uzunluklarını isteyecek.
- a veya b uzunlukları negatif veya 0 yazıldığında bu değerleri tekrar isteyecek.
- $c = \sqrt{a^2 + b^2}$ Alan= $a*b/2$, Çevre= $a+b+c$
- Bulunan değerler ekrana yazılacak.
- Program sonlanmadan önce bir tuşa basılana kadar beklenecek.

Çözüm yolu 1

- program ucgen;
- uses crt;
- var a,b,c,alan,cevre:real;
- begin
-
- clrscr;
-
- writeln('Bu program bir ABC dik üçgeninin hipotenüs uzunluğunu, alanını ve çevresini hesaplar...');
-
- repeat
- write('a uzunluğunu yazınız: ');
- readln(a);
- until a>0;

Çözüm yolu 1 - devamı

- repeat
- write('b uzunluğunu yazınız: ');
- readln(b);
- until b>0;
-
- $c := \sqrt{a^2 + b^2}$;
- $alan := a * b / 2$;
- $cevre := a + b + c$;
-
- writeln('Hipotenüs uzunluğu: ', c:4:2);
- writeln('Üçgen alanı: ', alan:4:2);
- writeln('Üçgen çevresi: ', cevre:4:2);
-
- readkey;
- end.

Çözüm yolu 2

c kenarı, alan ve çevre değişkenleri tanımlanmadan

- program soru17;
- uses crt;
- var a,b:real;
- begin
-
- clrscr;
-
- writeln('Bu program bir ABC dik üçgeninin hipotenüs uzunluğunu, alanını ve çevresini hesaplar...');
-
- repeat
- write('a uzunluğunu yazınız: ');
- readln(a);
- until a>0;

Çözüm yolu 2 - devamı

- repeat
- write('b uzunluğunu yazınız: ');
- readln(b);
- until b>0;

- writeln('Hipotenüs uzunluğu: ',sqrt(sqr(a)+sqr(b)):4:2);
- writeln('Üçgen alanı: ',a*b/2:4:2);
- writeln('Üçgen çevresi: ',a+b+(sqrt(sqr(a)+sqr(b))):4:2);

- readkey;
- end.

Örnek Soru

- Kullanıcı tarafından girilen iki sayı arasındaki sayıların kareleri toplamını hesaplayan ve ekranda gösteren programı yazınız.
- Program başlarken ekranı temizleyecek.
- Program kullanıcıdan x (küçük sayı) ve y (büyük sayı) sayılarını isteyecek.
- y sayısı x 'den küçük yazılırsa y sayısını tekrar isteyecek.
- Toplamı hesaplayıp ekrana yazılacak.
- Program sonlanmadan önce bir tuşa basılana kadar beklenecek.

Çözüm

- program karetoplam;
- uses crt;
- var x,y,i:integer;
- toplam:extended;
-
- begin
-
- clrscr;
-
- writeln('Bu program kullanıcının yazacağı iki sayı arasındaki sayıların');
- writeln('karelerinin toplamını yazar.');
-
- write('Küçük sayıyı yazınız... ');
- readln(x);

Çözüm - devamı

- repeat
- write('Büyük sayıyı yazınız..: ');
- readln(y);
- until (x<y);
-
- toplam:=0;
-
- for i:=x to y do toplam:=toplam+i*i;
-
- writeln('Toplam..: ',toplam);
-
- readkey;
- end.

Ödevler

- Faktöriyel alma programını **repeat-until** döngüsü kullanarak yazınız.
- Faktöriyel alma programını **while-do** döngüsü kullanarak yazınız.

Bazı Ekran Komutları

Ekran Komutu	Açıklama
clrscr;	Ekranı temizler.
Window (x1,y1,x2,y2);	Ekranın belirtilen koordinatlarında pencere çizer.
LowVideo;	Yazının parlaklık oranını düşürür.
NormVideo;	Yazının parlaklık oranını normale çevirir.
HighVideo;	Yazının parlaklık oranını yükseltir.
TextColor (Renk);	Yazı rengini değiştirir. (Renk 0 -15 arasındadır.)
TextBackGround(Renk);	Arka plan rengini değiştirir. (Renk 0 -15 arasındadır.)

Renk Değerleri

Renk Kodu	Renk	Renk Kodu	Renk
0	Siyah	8	Koyu Gri
1	Mavi	9	Açık Mavi
2	Yeşil	10	Açık Yeşil
3	Turkuaz	11	Açık Turkuaz
4	Kırmızı	12	Açık Kırmızı
5	Pembe	13	Açık Pembe
6	Kahverengi	14	Sarı
7	Parlak Gri	15	Beyaz

Matematiksel İşlem Komutları

Komut	Açıklama
Abs(x);	x sayısının mutlak değerini alır.
Sin(x);	x sayısının sinüs değerini verir. (Sonuç tipi : reel)
Cos(x);	x sayısının kosinüs değerini verir. (Sonuç tipi : reel)
ArcTan(x);	x sayısının ark tanjant değerini verir. (Sonuç tipi : reel)
Int(x);	x sayısının ondalık kısmını atar.
Frac(x);	x sayısının ondalık kısmını alıp, tam kısmını atar.
Exp(x);	x sayısının e üssü değerini verir.
Ln(x);	X sayısının doğal logaritma değerini verir.
Sqr(x);	x sayısının karesini alır.
Sqrt(x);	x sayısının karekökünü alır.
Round(x);	x ondalıklı sayısını kendisine en yakın tam sayıya yuvarlar.
Random(x);	0 ile x sayısı arasında rasgele sayı üretir.
Randomize;	Random komutu sıfırlar. random(x) dan önce kullanılır.
Trunc(x);	x sayısının tamsayı kısmını almak için kullanılır.

Örnek - Sinüs Teoremi

- Sinüs teoremi ile üçgen alanı hesaplayan program yazınız.
- Program kullanıcıya
 - a ve b kenar uzunluklarını ve
 - C açısının değerini
- Sorsun.
- Üçgen alanı ekrana yazdırılsın.

Çözüm - Sinüs Teoremi

- Program sinusteo;
- Uses crt;
- Var
- a,b,c:integer;
- alan : single;
- Begin
- Clrscr;
- Write('a kenarının uzunluğunu yazın: ');
- Readln (a);
- Write('b kenarının uzunluğunu yazın: ');
- Readln (b);
- Write('C açısını yazın: ');
- Readln (c);
- alan:=0.5*a*b*sin(c*pi/180);
- Writeln('üçgenin alanı = ',alan:4:2);
- Readkey;
- End.

Diziler

- Tek boyutlu dizi:

değişken_adı : **Array[X..Y] Of** değişken_tipi;

- Çok boyutlu dizi:

değişken_adı : **Array[X..Y,K..L] Of**
değişken_tipi;

Tek boyutlu dizi örneği – ortalama hesaplama

- Kullanıcının yazacağı “n” boyutlu dizide yer alan rakamların aritmetik ortalamasını hesaplayan program yazınız.

Aritmetik Ortalama (dizi örneği)

```
Program aritmetik_ortalama;
Uses crt;
Var dizi: array[1..100] of real;  n,sayac:integer;  toplam,aort:real;
begin
  Clrscr;
  toplam:=0;  aort:=0;
  Write('Lütfen dizi boyutunu (maks. 100) yazınız: ');  Readln(n);
  For sayac:=1 to n do
  begin
    write(sayac, ' . Elemanı yazınız: ');  readln(dizi[sayac]);
    toplam:=toplam+dizi[sayac];
  end;
  aort:=toplam/n;
  Writeln('Aritmetik ortalama: ',aort);
  Readkey;
end.
```

Ödev – geometrik ortalama hesaplama

- Kullanıcının yazacağı “n” boyutlu dizide yer alan rakamların geometrik ortalamasını hesaplayan program yazınız.
- Önemli not: Pascal’da kuvvet fonksiyonu bulunmamaktadır. Ancak x^y ‘yi hesaplamak için:

$$a = x^y$$

$$\ln(a) = \ln(x^y)$$

$$\ln(a) = y * \ln(x)$$

$$e^{\ln(a)} = e^{(y * \ln(x))}$$

$$a = x^y = e^{(y * \ln(x))}$$

Fonksiyonlar, Prosedürler, Veri dosyası okuma yazma

Doç. Dr. Özgür ZEYDAN

<https://ozgurzeydan.com.tr/>

Alt Programlar

- Program içerisinde sıkça tekrarlanan komutları azaltmak için bazı komutlar, alt programlar tarafından gerçekleştirilir.

Alt program türleri:

- Fonksiyon (Function)
- Alt Yordam (Prosedür - Procedure)
 - Parametrelili prosedürler
 - Parametresiz prosedürler

Fonksiyon Örneği

(İki sayıyı toplayan fonksiyon)

- program toplam_function;
- uses crt;
- var x,y,z:integer;
- **function Topla(sayi1,sayi2:integer):integer;**
- **begin**
- **Topla:=sayi1+sayi2;**
- **end;**
- begin
- clrscr;
- write('ilk sayıyı yaz: '); readln(x);
- write('ikinci sayıyı yaz: '); readln(y);
- **z:=Topla (x,y);**
- write('İki sayının toplamı: ',z);
- readkey;
- end.

Parametrelili Prosedür Örneği (İki sayıyı toplayan procedure)

- program toplam_procedure;
- uses crt;
- var x,y,z:integer;
- procedure Topla(sayi1,sayi2:integer; var sonuc:integer);
- begin
- sonuc:=sayi1+sayi2;
- end;
- begin
- clrscr;
- write('ilk sayıyı yaz: '); readln(x);
- write('ikinci sayıyı yaz: '); readln(y);
- Topla (x,y,z);
- write('İki sayının toplamı: ',z);
- readkey;
- end.

Parametrelili Prosedür Örneği (İki sayıyı toplayan procedure-2)

- program toplam_procedure;
- uses crt;
- var x,y,z:integer;
- procedure Topla(sayi1,sayi2:integer);
- begin
- z:=sayi1+sayi2;
- end;
- begin
- clrscr;
- write('ilk sayıyı yaz: '); readln(x);
- write('ikinci sayıyı yaz: '); readln(y);
- Topla (x,y);
- write('İki sayının toplamı: ',z);
- readkey;
- end.

Parametresiz Prosedür Örneği (faktöriyel hesabı)

- program faktoriyel_hesapla;
- uses crt;
- var n,i:integer;
- fakt:longint;
- procedure fhesapla;
- begin
- fakt:=1;
- for i:=1 to n do fakt:=fakt*i;
- end;
- begin
- clrscr;
- write('Faktöriyeli hesaplanacak sayıyı yazınız: ');
- readln(n);
- fhesapla;
- Writeln(n,' sayısının faktöriyeli: ',fakt:12);
- readkey;
- end.

Dosya Komutları

- `assign(dosya,'veri.txt');`
{dosya değişkenini veri.txt ile ilişkilendirir}
- `rewrite(dosya);`
{dosyayı yeniden oluşturur, eski veriler silinir}
- `append(dosya);`
{eski bilgileri koruyarak dosyayı yeniden yazmak için açar}
- `writeln(dosya,değişken_adı);`
{dosyaya yazar}
- `reset(dosya);`
{hazır dosyayı okumak için açar}

Dosya Komutları

- `readln(dosya,değişken_adı);`
{dosyadan okur}
- `erase(dosya);`
{dosyayı siler}
- `rename(dosya,yeniveri.txt);`
{dosyanın adını değiştirir}
- `close(dosya);`
{dosyayı kapatır}

TXT Dosyasına Veri Kaydetmek

- program txt_kaydet;
- uses crt;
- var **dosya:text**;
değişken_adı:değişken tipi;
- begin
- _____;
- **assign(dosya,'veri.txt');**
- **rewrite(dosya);**
- **writeln(dosya,değişken_adı);**
- **close(dosya);**
- _____;
- end.

TXT Dosyasından Veri Okumak

- program txt_kaydet;
- uses crt;
- var **dosya:text**;
değişken_adı:değişken tipi;
- begin
- _____;
- **assign(dosya,veri.txt);**
- **reset(dosya);**
- **readln(dosya,değişken_adı);**
- write(değişken_adı);
- _____;
- repeat
- _____;
- until (**EOF(dosya)**);
- end.

{EOF: End Of File}

Örnek – Dosya Programı

- 0 ile 100 arasındaki her bir °C değerinin fahrenheit karşılığını hesaplayıp “veriler.txt” isimli dosyaya kaydeden bir pascal programı yazınız.

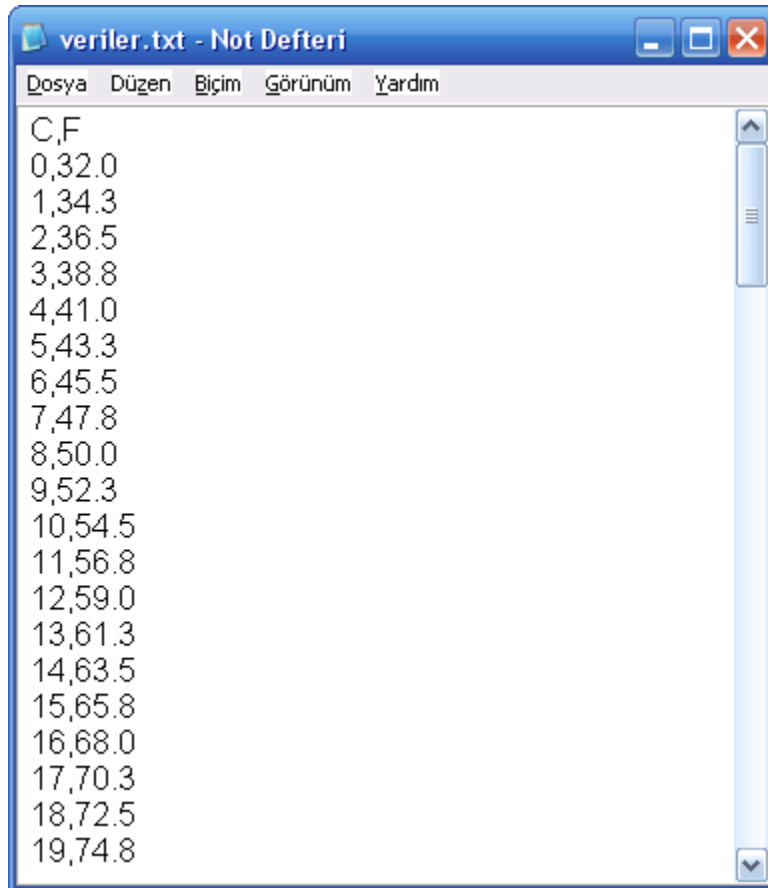
Örnek – Dosya Programı

- program cel_to_fah;
- uses crt;
- var c:byte;
- f:real;
- dosya:text;
- begin
- clrscr;
- writeln('Bu program 0 - 100 °C arasındaki değerlerin fahrenheit karşılığını hesaplar.');
- writeln('Daha sonra bu değerleri "veriler.txt" adıyla kaydeder.');
- writeln;
- writeln('Devam etmek için bir tuşa basınız...');
- readln;

Örnek – Dosya Programı

- `assign(dosya,'veriler.txt');`
- `rewrite(dosya);`
- `writeln(dosya,'C,F');`
- `for c:=0 to 100 do`
- `begin;`
- `f:=((9*c)/4)+32;`
- `writeln(dosya,c,',',f:3:1);`
- `end;`
- `close(dosya);`
- `writeln('C ve F değerleri "veriler.txt" dosyasına kaydedilmiştir.');`
- `writeln('Çıkış için bir tuşa basınız...');`
- `readln;`
- `end.`

Program Çıktı Dosyası



The screenshot shows a Notepad window with the title 'veriler.txt - Not Defteri'. The menu bar includes 'Dosya', 'Düzen', 'Biçim', 'Görünüm', and 'Yardım'. The text content is as follows:

```
C,F  
0,32.0  
1,34.3  
2,36.5  
3,38.8  
4,41.0  
5,43.3  
6,45.5  
7,47.8  
8,50.0  
9,52.3  
10,54.5  
11,56.8  
12,59.0  
13,61.3  
14,63.5  
15,65.8  
16,68.0  
17,70.3  
18,72.5  
19,74.8
```

- “veriler.txt” dosyası çalıştırılan pascal dosyası ile aynı klasör içindedir.