

FORTRAN'da Giriş/Çıkış Ve Tanımlamalar

- Giriş/Çıkış deyimleri
- Biçimli giriş/çıkış deyimleri,
- Klavye'den veri girişi
- Dosya'dan veri okuma ve dosyaya veri yazma
- Ekrana sonuçları yazdırma,
- Yazıcı, çizici gibi ortamlardan sonuç almak.

FORTRAN'da I/O Ve Tanımlamalar

- Serbest biçimde veri girişi/çıkışı,
- Biçimli veri girişi/çıkışı,
- Biçimli girişlerde kullanılan Edit tanımlayıcıları,

Giriş/çıkış için

Taşıyıcıların, sayıların, karakter, boşluk, tekrar belirteçlerinin kontrol edilmesi gerekir.

Veri Girişİ READ(k, n) liste

- Klavyeden (bilgisayara) veri girişi (bilgisayarın okuma yapabilmesi)
- Dosyadan (HDD, flash disk, CD/DVD, disket, gibi magnetik, elektrik veya optik ortamlardan) okuma
- Fare ile okuma vs.

READ(k, n) liste

Okuma deyiminde,

- k – ortamı
- n – biçimlendiricinin etiket numarasını
- k - * olursa klavyeyi
- n - * olursa serbest formatta girişi gösterir
- READ(4,*)

anlamı - 4 nolu ortamdan/birimden serbest okuma yapılacaktır

READ deyiminin genel kullanımı

```
READ(<dosya no> [,<format belirteci>][,REC=<kayıt numarası>][,END=<deyim numarası>][,ERR=<deyim numarası>]) <alan değişkenleri>
```

<dosya no>	okunan dosya numarası
<format belirteci>	FORMAT deyim numarası
REC=<kayıt numarası>	okunan dosyadaki kayıt numarası
END=<deyim numarası>	dosya sonu deyim numarası
ERR=<deyim numarası>	giriş çıkış hataları deyim numarasına aktarılır

READ(*, *) Birinci, İkinci, Üçüncü, Dördüncü

- Yukarıdaki satır **Birinci, İkinci, Üçüncü, Dördüncü** değişkenlerine serbest formatta bilgi aktarılacağını gösterir.

- `READ (1, 10, REC=5, END=20, ERR=30) A, B`
`10 FORMAT (F5.2, F7.3)`

Yukarıdaki satırlarda A ve B değişkenlerine 10 biçimleyici etiket numarasına göre A değişkenine F5.2 biçiminde, B değişkenine F7.3 biçiminde noktalı sayı aktarılacağı belirtilmektedir. Okumanın 1 nolu dosyanın (RECORD) 5nci kayıtlından olacağı, bu okuma esnasında dosya sonuna gelirse (END) 20 etiket numaralı satıra sapılacağı, okuma esnasında (ERROR) hata oluşursa 30 etiket numaralı satıra geçileceği belirtilmektedir.

WRITE(k,n) liste

Genel olarak çıkış aygıtlarından sonuç verilerinin alınması için kullanılır.

Sonuçların veya çıktıların yazıcıya aktarılabilmesi için k yerine * alınır ve program çalıştırılmadan önce klavyedeki Ctrl ve P tuşlarına aynı anda basılır (MS-DOS ortamında). Böylece program çalıştığı sürece ekrana yazılan her şey yazıcıya aktarılmış olur.

WRITE ifadesi biçimlendirilmiş veya biçimlendirilmemiş şekillerde kullanılabilir.

WRITE deyiminin genel yazımı

```
WRITE (<dosya no> [,<format belirteci>][,REC=<kayıt numarası>]  
[,ERR=<deyim numarası>]) <alan değişkenleri>
```

<dosya no>

okunan dosya numarası

<format belirteci>

FORMAT deyim numarası

REC=<kayıt
numarası>

okunan dosyadaki kayıt numarası

ERR=<deyim
numarası>

giriş çıkış hataları deyim numarasına
aktarılır

WRITE(*, *) A, B, C, D

- Yukarıdaki satır **A, B, C ve D** değişkenlerindeki bilgilerin serbest formatta ekrana aktarılacağını gösterir.

- `WRITE (1, 10, REC=5, ERR=30) A, B`
`10 FORMAT (F5.2, F7.3)`

Yukarıdaki satırlarda A ve B değişkenlerine 10 biçimleyici etiket numarasına göre A değişkenindeki değer F5.2 biçiminde, B değişkenindeki değer F7.3 biçiminde noktalı sayı olarak 1 nolu ortama aktarılacağını belirtilmektedir. Yazmanın 1 nolu dosyanın (RECORD) 5nci kaydına olacağını, yazma esnasında (ERROR) hata oluşursa 30 etiket numaralı satıra geçileceği belirtilmektedir.

OPEN deyimi

```
OPEN (<birim no>[, FILE = 'dosya_adı'][, STATUS='<durum>'][,  
ACCESS='<erişim>'][, FORM='<format>'][, RECL=<kayıt uzunluğu>])
```

<birim no>

açılan dosyanın FORTRAN daki numarası

FILE = 'dosya_adı'

Dosya adı kullanıma uygun olmalıdır

STATUS='<durum>'

durum-var olan dosyalar için OLD veya yeni
açılacak dosyalar için NEW

ACCESS='<erişim>'

erişim - SEQUENTIAL veya DIRECT olabilir

FORM='<format>'

Format - dosya kayıtlarının formatlı FORMATTED,
formatsız UNFORMATTED veya ikili sayı sisteminde
BINARY olarak açılmasını sağlar.

RECL=<kayıt uzunluğu>

Kayıt uzunluğu tam sayı olmalıdır.

OPEN deyimi

```
OPEN (1, FILE = 'veri.dat', STATUS='OLD', ACCESS='SEQUENTIAL',  
      FORM='FORMATTED', RECL=40)
```

Yukarıdaki deyim ile bilgisayarın 1 no'lu ortamında bulunan/var olan veri.dat isimli dosyanın her kayıt satırından, 40 karakter kayıt uzunluğu olan ve formatlı kayıt satırlarına sırayla ulaşılması/erişilmesi sağlanır. Kısaca program tarafından dosya kullanıma hazır hale getirilir.

OPEN deyimi

```
PROGRAM OrtalamaHiz
  IMPLICIT NONE
  LOGICAL :: Durum
  REAL :: x, t, v_ort, v_ani
  REAL :: x_ilk, t_ilk, x_son, t_son
  OPEN(1, FILE='konumzaman.txt')
5  READ(1, 10, END=15, ERR=20, IOSTAT=Durum) t, x
10  FORMAT(F3.1,1x,F4.1)
  WRITE(*,10) t, x
  GOTO 5
15  stop
END PROGRAM OrtalamaHiz
```

0.0	0.0
1.0	2.3
2.0	9.2
3.0	20.7
4.0	36.8
5.0	57.5

FORMAT (s1, s2, ... sk)

- **READ** ve **WRITE** deyimleri için verinin türü ve uzunluğunu bu deyim belirler. Burada n, ilgili **READ** veya **WRITE** deyiminde belirlenen **FORMAT** deyiminin numarasıdır. s'ler ise **READ** veya **WRITE** deyiminin **liste** kısmında sıralanan değişkenlerin türlerini, uzunluklarını ve yerlerini tanımlayan **FORMAT KODLARIDIR**.

Format Kodları

Veri türü	Kodu	Örnekler
Tam sayı	Iw	I5, I1
Tek duyarlı gerçel sayı	Fw.d	F6.2 , F5.0
Çift duyarlı gerçel sayı	Ew.d	E15.6, E9.2
Alfabetik karakter	Aw	A5, A8

Format Kodları

Veri	Format kodu	Çıktı
-12.34537	F9.3	bb-12.345
-12.34537	F9.2	bbb-12.35
20.52 x10 ⁻⁸	E12.4	bb0.2052E-06
20.52 x10 ⁻⁸	E12.6	0.205200E-06
20.52 x 10 ⁻⁸	E8.3	.205E-06
-0.152	E12.3	bb-0.152E+00
-0.152	1PE12.3	bb-1.520E-01
-0.152	2PE12.3	bb-15.20E-02
-0.152	2PE12.4	b-15.200E-02
-0. 152	G12.3	bb -0.152
0.1	G8.2	0.10
0.09	G8.2	0.90E-01
99	G8.2	B99
100	G8.2	0.10E+03

Format Kodları

Fonksiyonu	Kodu
Kolon atlama	nX
Satır atlama	/
Aynı satırda kal	\
Üç satır atlama	///
Yeni sayfaya geçiş	1H1

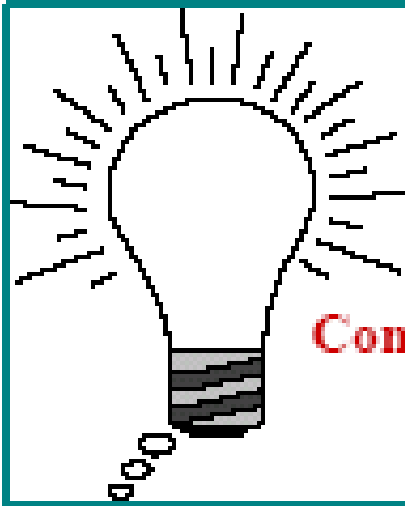
Format Kodları

<code>FORMAT (2XT3, 2X, I3, 2X, I3)</code>	yerine	<code>FORMAT (3 (2X, I3))</code>
<code>FORMAT (F5.2, F5.2, F5.2)</code>	yerine	<code>FORMAT (3F5.2)</code>
<code>WRITE (* , *) 'Basınç='</code>	yerine	<code>WRITE (* , *) P</code>
<code>WRITE (*, 10) P</code> <code>10 FORMAT (F6.2)</code>	yerine	<code>WRITE (*, 10) P</code> <code>10 FORMAT ('Basınç=', F6.2)</code>

Tanımlama Deyimleri

- Genellikle programın ilk satırlarında yer alırlar.
- Programda kullanılacak tüm tamsayı, gerçel ve kompleks sayı değişkenleri, alfabetik ve indisli değişkenler (diziler ve matrisler) ile mantıksal değişkenler, bu deyimler yardımıyla tanımlanmalıdır.
- Duyarlı hesaplamalarda kullanılacak değişkenlerin duyarlılığı da yine tanımlanmalıdır.

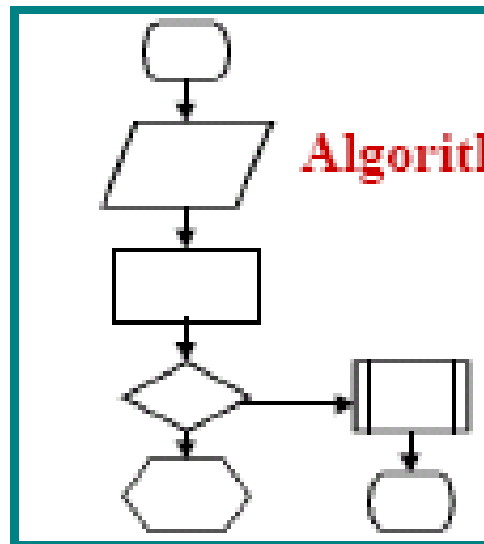
Programı hazırlama



Concept/Ideas

```
READ(*,*) A,B,C  
VECTOR=A*B  
IF(B>ABS(C)) THEN  
  B=C  
ELSE  
  A=C  
ENDIF  
WRITE(*,*)&  
&VECTOR,A,B
```

**Fortran90
Source Code**



Algorithm



Debugging

Tanımlama Deyimleri

- **DOUBLE PRECISION** : Noktalı sayı değişkenlerin, derleme ve hesaplama sırasında çift duyarlıklı olarak işlem görmelerini sağlar (Örnek : DOUBLE PRECISION HEAT, ENERJİ, A, B).
- **INTEGER** : Tam sayıları değişkenlerde saklamak için kullanılır (Örnek : INTEGER A,B,C).
- **REAL** : Noktalı sayı değişkenlerin, derleme ve hesaplama sırasında tek duyarlıklı olarak işlem görmelerini sağlar (Örnek : REAL ISI , MATRİS).
- **IMPLICIT** : Çok sayıda reel ve tam sayı değişkeninin tipini değiştirmede kullanılır. Tanımlama deyimlerinin en başında yer alır (Örnek: IMPLICIT REAL (I-N), IMPLICIT INTEGER (A-H)).
- **COMPLEX** : Programlarda kullanılacak kompleks sayı değişkenleri bu deyim yardımıyla tanımlanmalıdır (Örnek : COMPLEX * 5 CIM, BOM).
- **CHARACTER** : Programlarda alfabetik olarak kullanılacak değişkenler bu deyimle tanımlanır (Örnek : CHARACTER * 4 C, POINT * 5, A).
- **DIMENSION** : Bir programda kullanılacak dizi, matris gibi indisli değişkenlerin boyut ve eleman sayıları bu deyimle belirtilmelidir; böylece bellekte bunlar için gerekli yerlerin ayrılması da sağlanmış olur (Örnek : DIMENSION DİZİ(25), MATRİS(2:5)).
- **LOGICAL** : Bir programda kullanılacak mantıksal değişkenler bu deyimle belirtilmelidir. Mantıksal değişken I veya O değerini, mantıksal bir aktarma deyimini yardımıyla alabilir (Kesim 5.4; Örnek : LOGICAL TEST).

Soru 1

- Klavyeden READ deyimi ile girilen pozitif tam sayının tek veya çift sayı olup olmadığını IF deyimi ile sorgulayan ve “tek sayı” veya “çift sayı” şeklinde ekrana mesaj yazan ve 0 değeri girilince programın çalışması bitiren bir FORTRAN programını yazınız.

Çözüm 1

```
Program TekCift
Implicit none
Integer :: I
read (*,*) I
If((-1)** I == 1 ) write (*, *) I, ' çift sayıdır'
If((-1)** I == -1) write (*, *) I, ' tek sayıdır'
Stop
End program TekCift
```

Soru 2

Klavyeden READ deyimi ile A, B, C, D deęişkenlerine noktalı sayı deęerlerini okuyan ve yine klavyeden girilen J tamsayı deęişkeninin alacaęı deęere göre ařaęıdaki hesaplamaları yapan FORTRAN programını yazınız.

	A	B	C	D	J
J=1 ise $X=A+B+C+D$	1.0	2.0	30	4.0	1
J=2 ise $X=(A+B+C+D)/4.0$	5.6	2.0	10.4	4.0	2
J=3 ise $X=(A+B)/(C+D)$	-10.5	3.5	25.2	6.3	3
J=4 ise $X=A/B+C/D$	27.8	38.33	45.75	-41.02	4
J=5 ise $X=A/B-C/D$	33.68	21.46	13.21	14.36	5

Çözüm 2

```
Program Fonksiyon
Implicit none
Real :: A, B, C, D, X
Integer :: J
Read (*,*) A, B, C, D
read (*,*) J
If (J == 1 ) X=A+B+C+D
If (J == 2 ) X=(A+B+C+D)/4.0
If (J == 3 ) X=(A+B)/(C+D)
If (J == 4 ) X=A/B+C/D
If (J == 5 ) X=A/B-C/D
write (*, *) 'seçiminize göre sonuç ', X
Stop
End program Fonksiyon
```


Soru 3

- 3. Bir mil tarafından aktarılan dönme momentini (torku) (lb.in) beygir gücüne (hp) ve milin dönme sayısına (rpm) bağlı olarak

- $$T = \frac{hp \times 63000}{rpm}$$

- denklemleri ile hesaplayan ve buradan bu torku sağlayacak milin çapı :

$$D = \sqrt[3]{\frac{16T}{\pi S}}$$

- Denklemdaki S birim yüzeye uygulanan basınç olarak (lb/in²). Klavyeden girilen beygir gücüne (hp), dakikadaki dönme sayısına (rpm) ve uygulanan baskıya (S) göre milin çapını belirleyen bir FORTRAN programını yazınız (S=8000 lb/in²).

Çözüm 3

```
Program BeygirGucu
Implicit none
Real :: hp, Sabit = 63000.0, rpm = 2000.0, S, T, DS = 8000.0
Read (*, *) hp
T = hp * Sabit/rpm
D = (16.0*T/(3.14159*S))**1/3
write (*, *) ' Bir mil ',hp, ' beygir gücü ile dakikada ',rpm, ' döndürülerek'
write (*, *) ' üretilen tork = ', T, ' lb.in tir. '
write (*, *) ' Bu torku elde etmek için '
Write (*, *) ' milin çapı en az = ', D, ' inch olmalıdır.'
Stop
End program BeygirGucu
```

Soru 4

- DİE nin her ay belirlediği aylık TEFE oranları (%) olarak aşağıdaki çizelgede verilmektedir. Bu verileri bir dosyadan okuyup ekrana aşağıdaki formattaki gibi yazan programı hazırlayınız.

Yıllar	1995	1996	1997	1998	1999	2000	2001	2002
Ocak	10.3	5.8	5.6	6.5	3.6	5.8	2.3	4.2
Şubat	7.1	4.1	6.2	4.6	3.4	4.1	2.6	2.6
Mart	5.5	7.0	6.0	4.0	4.0	3.1	10.1	1.9
Nisan	5.1	8.1	5.5	4.0	5.3	2.4	14.4	1.8
Mayıs	1.9	4.1	5.2	3.3	3.2	1.7	6.3	0.4
Haziran	1.3	2.7	3.4	1.6	1.8	0.3	2.9	1.2
Temmuz	2.3	2.4	5.3	2.5	4.0	1.0	3.3	2.7
Ağustos	2.3	3.8	5.8	2.4	3.3	0.9	3.5	2.1
Eylül	4.6	5.1	6.3	5.3	5.9	2.3	5.4	3.1
Ekim	3.7	5.5	6.7	4.1	4.7	2.8	6.7	3.1
Kasım	3.5	5.1	5.6	3.4	4.1	2.4	4.2	1.6
Aralık	4.4	3.9	5.4	2.5	6.8	1.9	4.1	2.6
Yıllık TEFE	52.0	57.6	67.0	44.2	50.1	28.7	65.8	27.3