

C Programlama

Öğretim Elemanı Bilgileri

- Dr. Öğr. Üyesi Sema ATASEVER
- Nevşehir Hacı Bektaş Veli Üniversitesi, Mühendislik Mim.Fak. Bilgisayar Mühendisliği
- Web sayfası : <https://biz.nevsehir.edu.tr/sema/tr>
- Email : sema@nevsehir.edu.tr | s.atasever@gmail.com

Ölçme Yöntemi

ARA SINAV

- Ara sınav : 100 puan üzerinden değerlendirilecektir , Katkı : %40

FİNAL ÖDEVİ

- Final Sınavı : 100 puan üzerinden değerlendirilecektir , Katkı : %60
- Nihai ders notu hesabı : Ara sınavın %40'ı, Final notunun %60'ı alınarak hesaplanmaktadır!

2.Hafta Konuları

- Algoritma ve akış diyagramı kavramları, veri türleri, değişken kullanımı.

Haftanın Sorusu ?

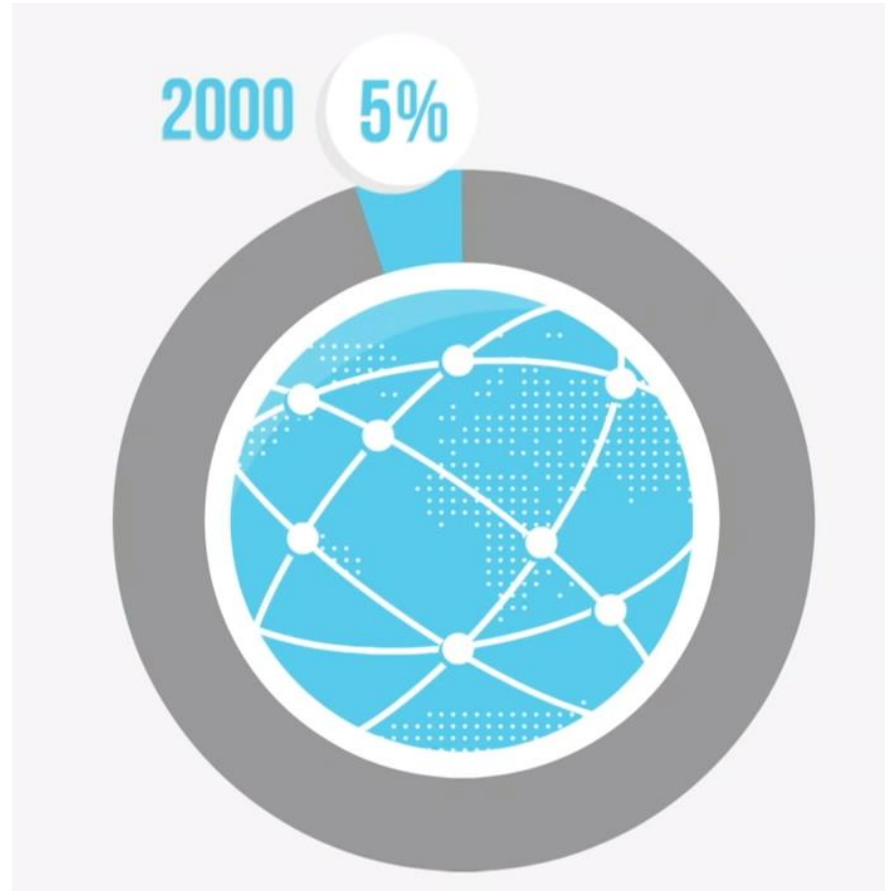


[www.menti.com](https://www.menti.com/79egsyu5pb)

Kod: 3273841

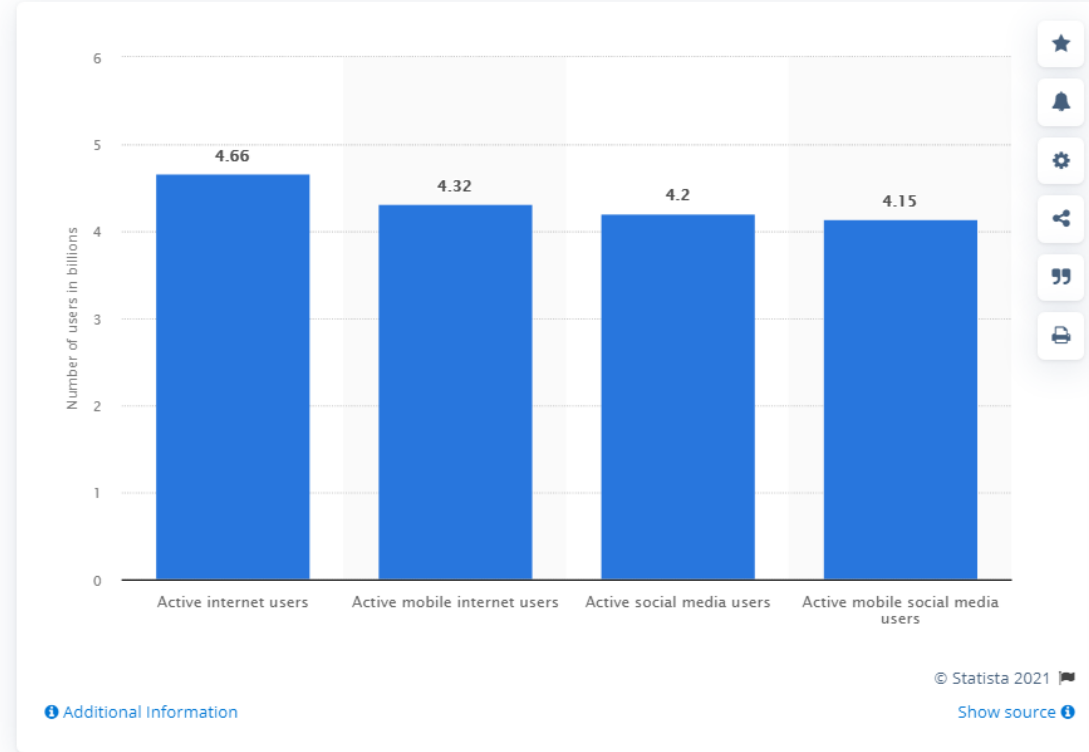
<https://www.menti.com/79egsyu5pb>

www Kullanımı



Ocak 2021 itibarıyla küresel dijital nüfus (milyar olarak)

Global digital population as of January 2021
(in billions)

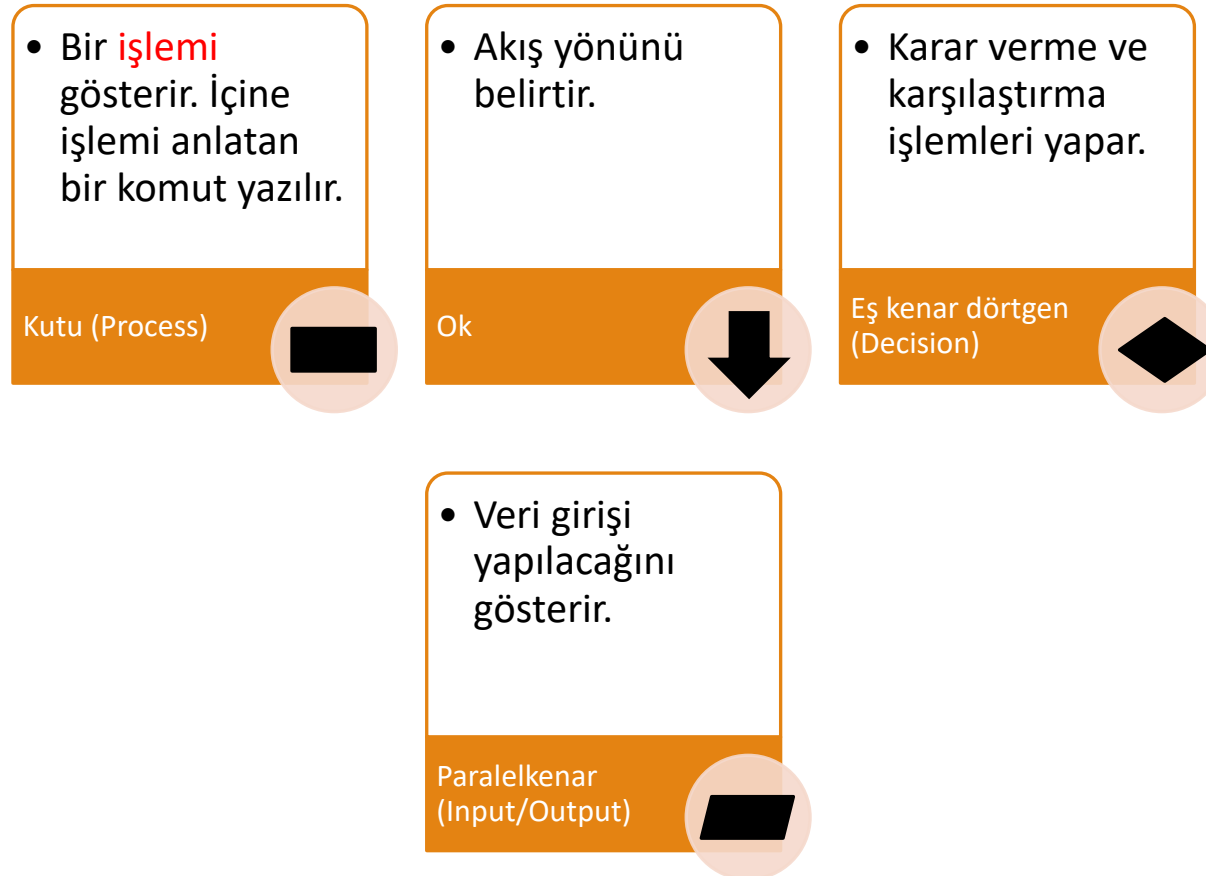


Tim Berners-Lee: A Magna Carta for the web

- <https://www.youtube.com/watch?v=rCplocVemjo> veya <https://cy.tc/O0zI>



Akış Diyagramı



- Algoritmaları göstermek için sıkça kullanılan yöntemlerden biri akış diyagramlarıdır.

Akış Diyagramı

Tamamı

- Köşeleri yuvarlatılmış, başla
- Köşeleri yuvarlatılmış, dur

Bir kısmı

- Boş yuvarlak ile başlar
- Boş yuvarlak ile biter

Örnek : En büyük eleman bulma

50 öğrencili bir sınıfta en yüksek notun bulunması algoritması:

1. Dizideki ilk notu en yüksek not olarak seç ve sırayı ikinci öğrenciye geçir.
2. Sırada öğrenci varsa 3. adıma, yoksa 5. adıma git.
3. Sıradaki öğrencinin notu şu ana kadarki en yüksek nottan büyükse bu yeni notu en yüksek not olarak seç.
4. Sırayı bir sonraki öğrenciye geçir ve 2. adıma dön.
5. En yüksek notu bildir.

Örnek : En büyük eleman bulma

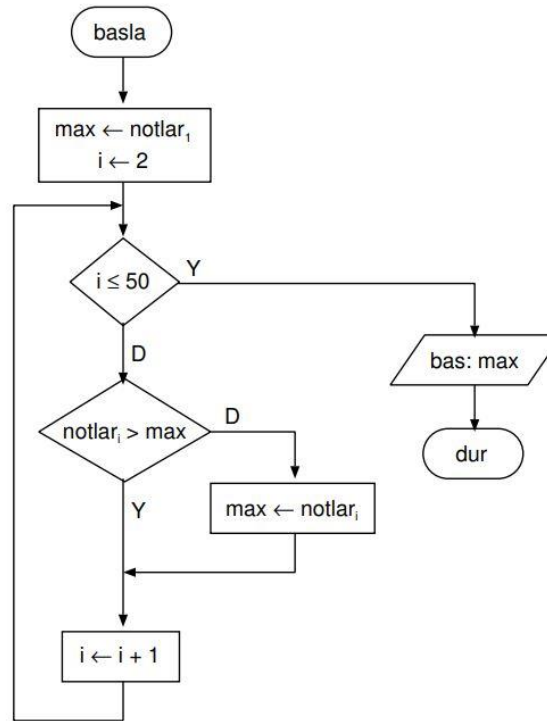
50 öğrencili bir sınıfta en yüksek notun bulunması algoritması:

1. Dizideki ilk notu en yüksek not olarak seç ve sırayı ikinci öğrenciye geçir.
2. Sırada öğrenci varsa 3. adıma, yoksa 5. adıma git.
3. Sıradaki öğrencinin notu şu ana kadarki en yüksek nottan büyükse bu yeni notu en yüksek not olarak seç.
4. Sırayı bir sonraki öğrenciye geçir ve 2. adıma dön.
5. En yüksek notu bildir.

Örnek : En büyük eleman bulma

1. $\text{max} \leftarrow \text{notlar}_1, i \leftarrow 2$
2. $i \leq 50$ ise 3. adıma, değilse 5. adıma git.
3. $\text{notlar}_i > \text{max}$ ise $\text{max} \leftarrow \text{notlar}_i$
4. $i \leftarrow i + 1$ ve 2. adıma dön.
5. en yüksek not: max

Örnek : En büyük eleman bulma



Şekil 1.11: En büyük eleman bulma algoritmasının akış çizeneği.

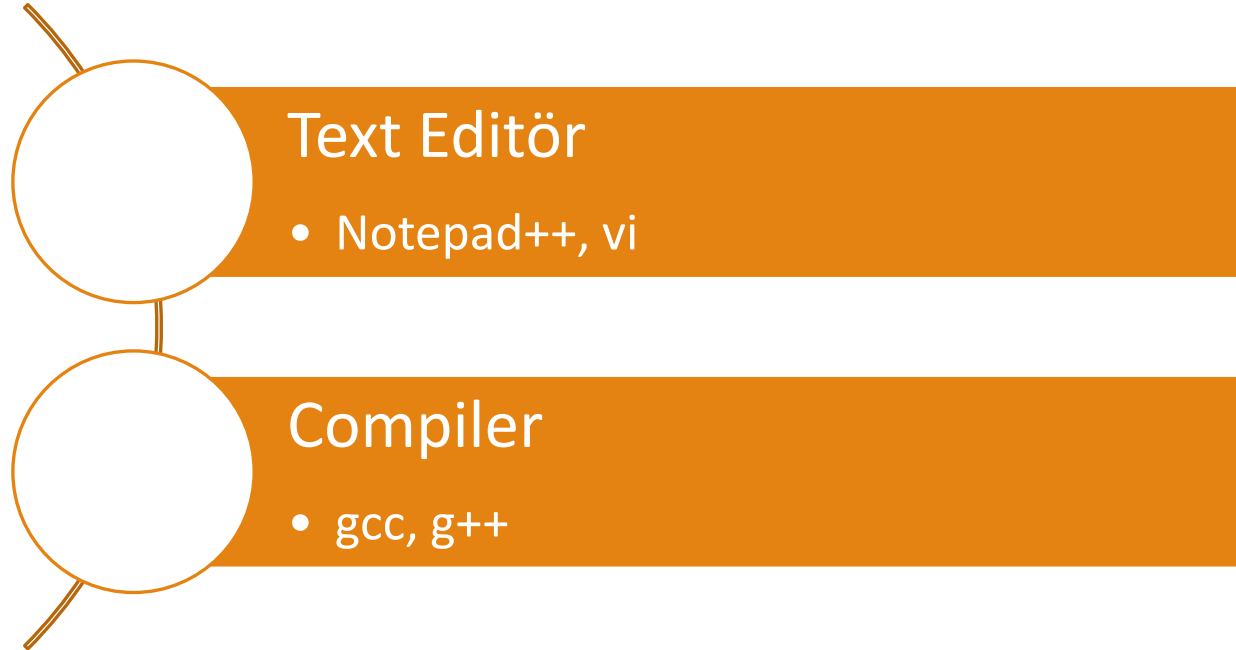
Örnek : En büyük eleman bulma

max	i	$i \leq 6$	notlar _i > max
43	2	D (2 < 6)	D (74 > 43)
74	3	D (3 < 6)	Y (65 < 74)
	4	D (4 < 6)	Y (58 < 74)
	5	D (5 < 6)	D (82 > 74)
82	6	D (6 = 6)	Y (37 < 82)
	7	Y (7 > 6)	

- 50 eleman yerine 6 elemanlı bir dizide en büyük elemanın bulunması algoritmasının işleyişi Tablo 1.1'de verilmiştir.

Tablo 1.1: En büyük eleman bulma algoritmasının örnek değerlerle işleyişi.

Kod yazmak için ihtiyacımız olanlar

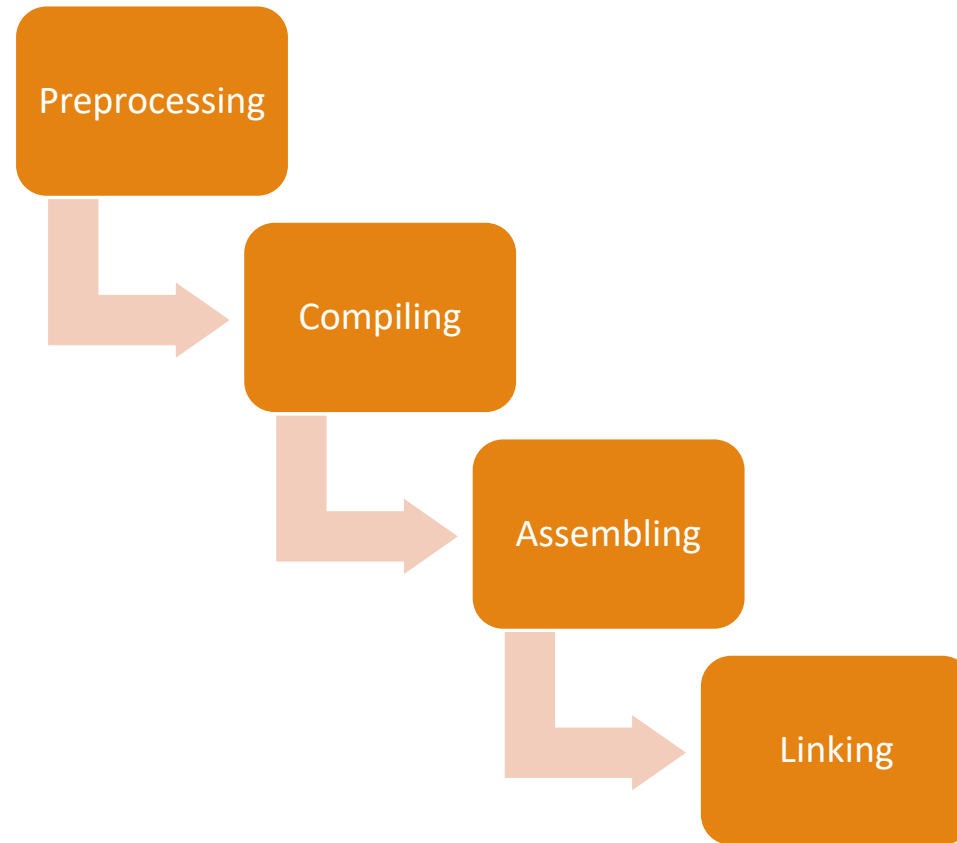


C Programı Çalıştırma

- Tüm C programları, C kodunu bilgisayar tarafından okunabilen makine koduna çevirmek için bir derleyici (Compiler) kullanılmalıdır. Yani bir c programını her çalıştırmak istediğinizde önce onu derlemeniz gerekir.
- Derleme kaynak kodundan (source code) makine koduna bir soyutlama işlemidir.



Derleme Adımları



Preprocessing (Ön işlem)

`#include <stdio.h>` *// Ön işlemci direktifleri, # işareti derleyiciye bu satırların önce işlenmesini söyler*

```
int main() {  
    // printf() displays the string inside quotation  
    printf("Hello, World!");  
    return 0;  
}
```

- Kütüphanelerin içerisinde prototip dediğimiz kod satırları yer almaktadır.
- `#include` : içeriğin bulunması ve değiştirilmesidir.

Compiling (Derleme)

- Assembly kodları (işlemcinin anladığı alt seviye komutlar). x86 64 bit çevirme komutları.

```
...
main:
    .cfi_startproc
# BB#0:
    pushq    %rbp
.Ltmp0:
    .cfi_def_cfa_offset 16
.Ltmp1:
    .cfi_offset %rbp, -16
    movq     %rsp, %rbp
.Ltmp2:
    .cfi_def_cfa_register %rbp
    subq     $16, %rsp
    xorl     %eax, %eax
    movl     %eax, %edi
    movabsq  $.L.str, %rsi
    movb     $0, %al
    callq    get_string
    movabsq  $.L.str.1, %rdi
    movq     %rax, -8(%rbp)
    movq     -8(%rbp), %rsi
    movb     $0, %al
    callq    printf
...
```

Assembling (Çeviri)

- Assembly dilinin 0 ve 1 lere (makine koduna) çevrildiği adım.

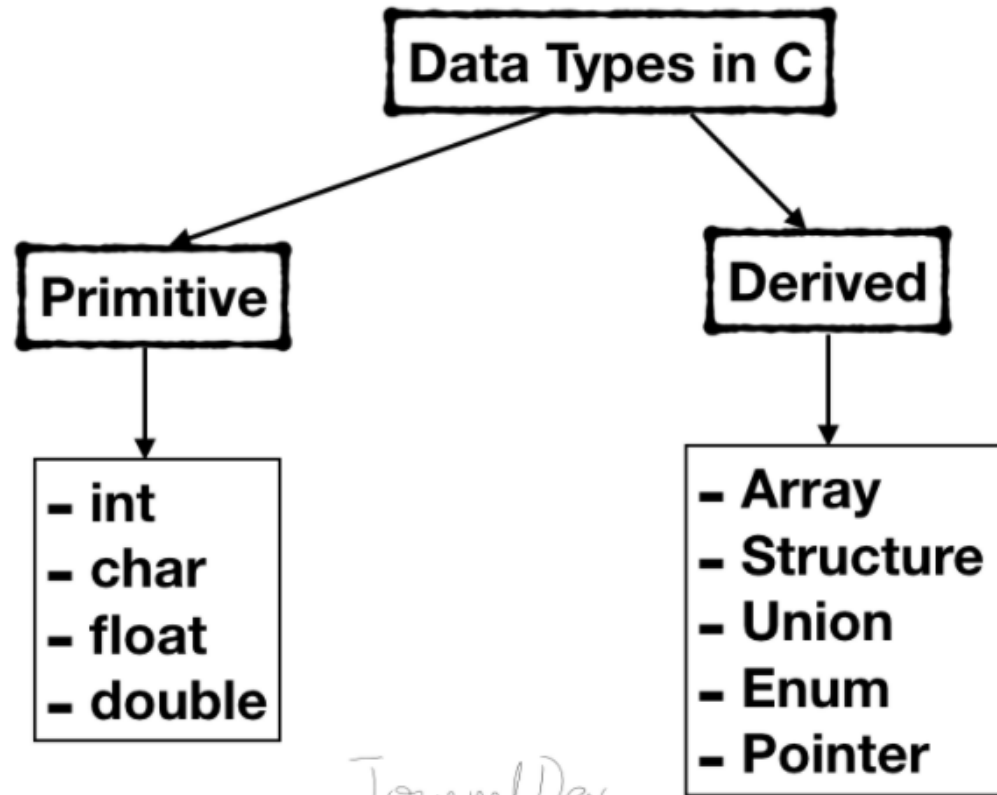
```
01111111010001010100110001000110
00000010000000010000000100000000
00000000000000000000000000000000
00000000000000000000000000000000
00000001000000000011110000000000
00000001000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
10100000000001000000000000000000
00000000000000000000000000000000
00000000000000000000000000000000
01000000000000000000000000000000
00000000000000000100000000000000
0000101000000000000000100000000
01010101010010001000100111100101
0100100010000011110110000010000
00110001110000001000100111000111
01001000101111000000000000000000
00000000000000000000000000000000
00000000000000000101100000000000
11101000000000000000000000000000
0000000001001000101111100000000
00000000000000000000000000000000
0000000000000000000000001001000
...
```

Linking (Bağlama)

- #include komutu ile programa dahil edilen diğer kodların ve programcının yazdığı kodların birleştirildiği adımdır.

[illegible][illegible][illegible]

Veri Türleri



JournalDev

1. Birincil Veri Türleri (Primitive)

1. int

Tamsayı değerlerini saklamak için kullanılır. GCC derleyicisi (32-Bit) ile derlenen C programı -2147483648 ile 2147483647 arasındaki tamsayıları depolayabilir. Int'in boyutu derleyiciye bağlıdır. GCC gibi 32 bitlik bir derleyicide hafızada 4 bayt'lık yer kaplar.

Örnek:

```
int yas = 18;
```


1. Birincil Veri Türleri (Primitive)

2. char

Sayı, sembol veya özel karakter dahil olmak üzere "a", "Z", "@" vb. Gibi tek bir karakteri depolar. Her karakterin saklanması hafızada 1 bayt (8 bit) yer kaplar.

Not : Her karaktere karşılık gelen bir ASCII değeri vardır. Örneğin, karakter olarak "1", ASCII değeri 49'a, "A" ise ASCII değeri olarak 65'e karşılık gelmektedir.

Örnek:

```
char cinsiyet = 'K';
```

Ödev2

- Kullanıcı **-1** girinceye kadar girilen sayıların toplamını hesaplayıp sonucu ekrana yazdıran **Algoritmayı** yazınız ve bu algoritmaya karşılık gelen **Akış Diyagramını** çiziniz.