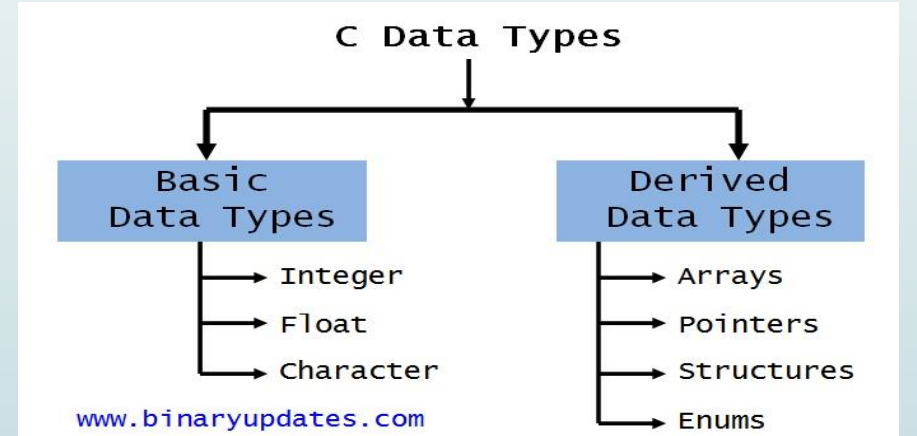


DEĞİŞKENLER VE VERİ TİPLERİ

Hazırlayan : Muhammed Esad Goncaloğlu

VERİ TİPLERİ

- Bir programda farklı veri tipleriyle işlem yapmamız gerekebilir.
- Örneğin, tamsayılar, kesirli sayılar, karakterler (harfler ve klavyedeki diğer simgeler), metinler (string), mantıksal (boolean) değerler (doğru=true, yanlış=false) ilk aklımıza gelen farklı veri tipleridir.
- Değişken tanımlarken onun hangi tip veriyi tutacağını belirtir.
- Böylece, ana bellekte o değişkene yetecek bir yer ayırır ve o veri tipine uygun işlemlerin yapılmasına izin verir.



Type	Size (bits)	Size (bytes)	Range
char	8	1	-128 to 127
unsigned char	8	1	0 to 255
int	16	2	-2^{15} to $2^{15}-1$
unsigned int	16	2	0 to $2^{16}-1$
short int	8	1	-128 to 127
unsigned short int	8	1	0 to 255
long int	32	4	-2^{31} to $2^{31}-1$
unsigned long int	32	4	0 to $2^{32}-1$
float	32	4	$3.4\text{E}-38$ to $3.4\text{E}+38$
double	64	8	$1.7\text{E}-308$ to $1.7\text{E}+308$
long double	80	10	$3.4\text{E}-4932$ to $1.1\text{E}+4932$

DEĞİŞKEN KAVRAMI

- Değişkenler bir programlama dilinde verilerin depolanma alanlarını temsil eder.
- Tanımlanan her değişkene bellek bölgesinden bir alan ayrılır.
- Bu bellek bölgesine okuma ve yazma işlemleri ise değişken ismi üzerinden sağlanır.
- Genel olarak değişkenler yandaki şekildeki gibi tanımlanır.

<veri tipi> <değişken adı>;

```
#include<stdio.h>
int main()
{
    int num1=1;
    int num2=2;
    if(num1<num2) //test-condition
    {
        printf("num1 is smaller than num2");
    }
    return 0;
}
```

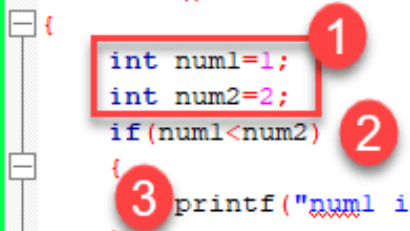
Diagram illustrating variable declaration and usage in C code:

- 1: Variable declaration (`int num1=1;` and `int num2=2;`)
- 2: Test condition (`if(num1<num2)`)
- 3: Action/Statement (`printf("num1 is smaller than num2");`)

DEĞİŞKENLERİN ÖZELLİKLERİ

- Bir değişkene değer atama işlemi tanımlarken yapılabilir.
- Bir değişkene değer atama işlemi yandaki örnekte olduğu gibi program içinde herhangi bir satırda yapılabilir.
- Bir veri tipi altında birden fazla isimle farklı değişkenler tanımlanabilir.

```
#include<stdio.h>
int main()
{
    int num1=1;
    int num2=2;
    if(num1<num2) //test-condition
    {
        printf("num1 is smaller than num2");
    }
    return 0;
}
```



```
int *p1,*p2,*p3;
p1=s1;
p2=s2;
p3=s3;
```

İSİMLENDİRME KURALLARI

- Değişkenlerin isimleri alfabede bulunan karakterlerle veya _(alt çizgi) ile başlamalıdır. Ama ilk harf hariç diğer karakterler sayı olabilir.
- Bazı programlama dilleri büyük ve küçük harf duyarlıdır. Yani Sayı, sayı ve SAYI hepsi ayrı değişken olarak algınalır.
- Değişken isimleri birden fazla kelime olduğu zaman; kelimelerin arasına boşluk konmaz. Bu tür değişkenleri ya kelimeleri birleştirerek veya kelimeler arasına _(alt çizgi) karakteri koyarak isimlendiririz.
- Değişkenlerin isimleri !, ?, {, } gibi karakterler içeremez.
- Programlama dili için tanımlanmış anahtar kelimelerini de değişken isimleri olarak kullanamayız.

Identify which of the following are valid variable names

roll#	✗	a	✓
roll_number_250	✓	i	✓
_volume	✓	p	✓
integer	✓	do_while	✓
Float	✓	string	✓
goto	✗	character	✓
Go_to	✗	post@home	✗
15_August_2020	✗	y_2020	✓
August-15	✗	bus	✓
pecific_ocean	✓		

-Dipakkumar Patel

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

DEĞİŞKENLERİ YAZDIRMA

- Değişkenleri yazdırırken hangi veri tipinde tanımlandıysa, o veri tipinin belirticisi ile yazdırılır.
- Yandaki görselde bu belirticileri veri tiplerine göre sıralanmıştır.
- Alttaki örnekte görüldüğü üzere değişkenler veri tiplerinin göre belirticileri ile printf() fonksiyonu vasıtasıyla yazdırılır.

Data type		Format specifier
Integer	short signed	%d or %i
	short unsigned	%u
	long signed	%ld
	long unsigned	%lu
	unsigned hexadecimal	%x
	unsigned octal	%o
Real	float	%f
	double	%lf
Character	signed character	%c
	unsigned character	%c
String		%s


```
int i = 10;
```

```
char c = 'a';
```

```
printf("Value of i = %d\n",i);
```

```
printf("Value of c = %c\n",c);
```

```
printf("i = %d \t c = %c \n",i,c);
```



KAYNAKÇA

- <https://binaryupdates.com/wp-content/uploads/Variables-and-Data-Types-in-C.jpg>
- <http://startertutorials.com/blog/wp-content/uploads/2016/04/data-types-sizes-in-c.png>
- https://www.guru99.com/images/1/020819_0436_CConditiona1.png
- <https://www.studyelectronics.in/wp-content/uploads/2018/06/Screenshot-719.png>
- <https://media.geeksforgeeks.org/wp-content/uploads/20221202181520/Cvariables2.png>
- https://i.ytimg.com/vi/ARGO_IU1H4k/mqdefault.jpg
- <https://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/java/ch05/dataTypes.htm#:~:text=Bir%20programda%20farkl%C4%B1%20veri%20tipleriyle,akl%C4%B1m%C4%B1za%20gelen%20farkl%C4%B1%20veri%20tipleri%20dir.>
- <https://aticleworld.com/character-set-and-keywords-in-c/>
- <http://c-programmingbooks.blogspot.com/2011/11/format-specifications-in-c-programming.html>
- https://i.ytimg.com/vi/nnnFM_NTSEM/maxresdefault.jpg