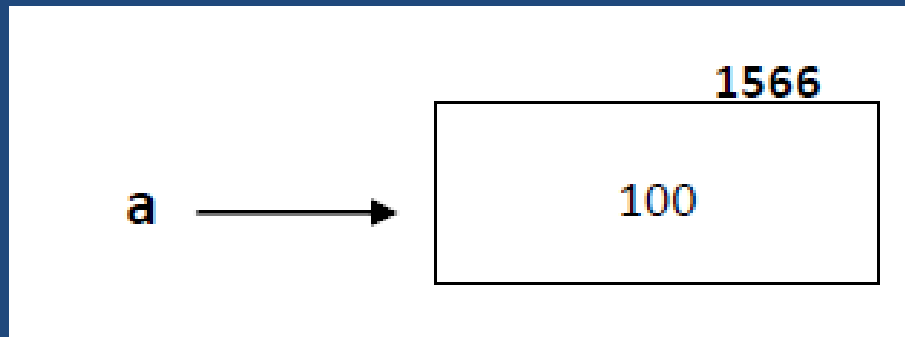


Problem Solving Through programming in C Course Code:ONL1001 Variable and data type

Ms. Shubhra dwivedi
Assistant Professor
School - SCOPE
VIT-AP Amaravati
Shubhra.d@vitap.ac.in

Variables

- Variable is simply name given to memory location which acts as placeholder or container for storing data. It may help to think of variables as a container for a value.

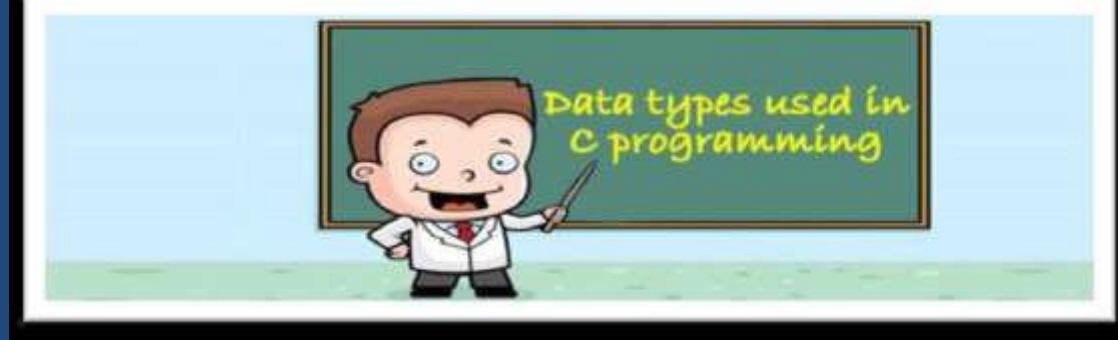


Data type

- We can store variables in our computer memory. Since each type of data takes different amount of memory, like integer takes 2 bytes, decimal numbers take 4 byte, and SINGLE character takes 1 byte. So we have to tell computer about the type of data we are going to store in our variable.

A Data Type Is Used to-

- Identify the type of a variable when the variable is declared.
- Identify the type of the return value of a function.
- Identify the type of a Parameter expected by a function.



- **Predefined/inbuilt :**
These datatypes are inbuilt.
eg : char, int, float, double
- **Drived Datatype**
These datatypes are inherited from predefined datatype. eg : arrays, pointer
- **Userdefined datatype :** These are created by user for his/her own purpose
- eg : typedef, enum, structure, union

Data type

```
graph TD; DT[Data type] --> P[Primitive]; DT --> D[Derived]; DT --> UD[User defined]; P --> P_list["char<br/>int<br/>float<br/>double<br/>void"]; D --> D_list["Array<br/>Pointer<br/>Function"]; UD --> UD_list["enum<br/>Structure<br/>Union"];
```

Primitive

char

int

float

double

void

Derived

Array

Pointer

Function

User defined

enum

Structure

Union

Predefined Datatypes

Type	Keyword
Boolean	Bool
Character	Char
Integer	Int
Floating point	Float
Double floating point	Double
Valueless	Void
Wide character	wchar_t

Basic Types

Integer types

- Integers are whole numbers with the wide range of values that are machine dependent.
- Keyword `int` is used for declaring the variable with integer type. For example--

`int var1;`

- Integer occupies 2 bytes memory space and its value range limited to -32768 to 32767
- Range of integer is 2^{-15} to 2^{+15}
- Each type is again classified into signed and unsigned integer.

- The following table provides the details of standard integer types with their storage sizes and value ranges-

Type	Storage size	Value range
Int	2 bytes	-32,768 to 32,767
unsigned int	2 bytes	0 to 65,535
short	2 bytes	-32,768 to 32,767
Unsigned Short	2 bytes	0 to 65,535
Long	4 bytes	-2,147,483,648 to 2,147,483,647
Unsigned long	4 bytes	0 to 4,294,967,295

Character types

- All single character used in programs belong to character type.
- The range of values that can be stored in a variable of character data type is -128 to 127.
- The char data type holds exactly 8bits (1 byte).
- Keyword char is used for declaring the variable with character type. For example--

char var1=h;

Here, var1 is a variable of type character which is storing a character 'h'.



CONTINUED...

Unsigned Char-

The unsigned char is a variation of char type. The size of a variable of unsigned char is also 1 byte. As the name itself indicates, the range of values that can be stored in a variable of type unsigned char is “0 to 255”.

Floating-Point Type

- All numeric data type items with fractional part belong to float type.
- The keyword float is used to declare variables of float type.

float var1;

- A variable of float type requires 4 bytes and the range of values that can be stored in it, is 3.4×10^{-38} to 3.4×10^{38}

•The following table provide the details of standard floating-point types with storage sizes and value ranges and their precision-

TYPE	STORAGE SIZE	VALUE RANGE	PRECISION
Float	4 byte	1.2E-38 to 3.4E+38	6 decimal places
Double	8 byte	2.3E-308 to 1.7E+308	15 decimal places
Long double	10 byte	3.4E-4932 to 1.1E+4932	19 decimal places

The Type `void`

- The void type specifies that no value is available. It is used in two kinds of situations-

<i>S.N.</i>	<i>TYPES & DESCRIPTION</i>
1.	Function returns as void There are various functions in C which do not return any value or you can say they return void. A function with no return value has the return type as void.
2.	Function arguments as void There are various functions in C which do not accept any parameter. A function with no parameter can accept a void.

Identifiers or variable

- Identifier is simply a name given to your variable, class, union, structure, function, etc.
- **Rules for creating valid identifiers:**
 - 1. An identifier can be any combination of alphabets, digits, underscore.
 - 2. Neither spaces nor special symbol other than underscore can be used.
 - 3. It won't begin with a digit

Valid and Invalid identifiers

Valid Identifiers	Invalid Identifiers
A_	9a
_a	A_B
A56	5a6
Hello	Hello?
Hello_hi	Hello hi
A9b	A9b\$

Declaration and initialization of variable

- **How to declare variable?**
- Declaration of variable means specify data type of variable and name assigned (**identifier**).
- A variable declaration provides assurance to the compiler that there is one variable existing with the given type and name so that compiler proceed for further compilation without needing complete detail about the variable.

- syntax :

datatype variable_name;

eg :

int a;

float b;

char c;

double first;

int i, j, k;

Initialization

- It means assigning the values to variable

```
int a;
```

```
a = 10;
```

```
float b;
```

```
b = 30.05;
```

```
char c;
```

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

Declaration and initialization in single step

```
int a = 10;
```

```
float b = 30.05;
```

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

(Value to characters always assigned in single quotes)

Keywords

- Are simply the reserved words whose meaning has been explained in library of C language.
- There are 32 keywords in c language.
- **We can't use keywords as identifiers or variables** because if we do so we are trying to assign a new meaning to the keyword, which is not allowed by the computer

List of Keywords

Auto	Goto	Struct	For
double	sizeof	break	signed
int	volatile	else	void
union	do	long	default
const	if	switch	extern
float	static	case	return
short	continue	enum	typedef
unsigned	while	register	char

```
int float;
```

```
charif;
```

```
Int void;
```

Above will give **error** as we are using keywords as identifier or variable name

Constants

- Is an identity which does not change its value. Once the value is initialized, that can't be changed.
- Constants are just like variables only condition is that their value doesn't change.
- We will use **const** keyword for declaring constants.

Syntax:

```
const datatype    variable_name = value;  
const int a = 10;
```

Thank You