



S4_2 Data Types in C



Learning objectives

To learn and appreciate the following concepts

- ✓ Data types in C
- ✓ Variable declaration

Learning Outcomes

At the end of session the student will be able to

- ✓ Understand different types of basic data types available in C
- ✓ Declare variables in a C program and use

The character type **char**

- A **char** variable can be used to store a single character.
- A **character constant** is formed by enclosing the character within a pair of single quotation marks. Valid examples: `'a'`.
- Character zero (`'0'`) is not the same as the number (integer constant) 0.
- The character constant `'\n'`—the newline character—is a valid character constant. It is called an **escape character**.
- There are other **escape sequences** like, `\t` for tab, `\v` for vertical tab etc.

Character Types

Character type **char** is related to the integer type.

- Modifiers (type specifiers) ***unsigned*** and ***signed*** can be used
 - **char → 1 byte (-128 to 127)**
 - **signed char → 1 byte (-128 to 127)**
 - **unsigned char → 1 byte (0 to 255)**
- **ASCII** (American Standard Code for Information Interchange) is the dominant encoding scheme for characters.
 - **Examples**

✓ ' ' encoded as 32	'+' encoded as 43
✓ 'A' encoded as 65	'Z' encoded as 90
✓ 'a' encoded as 97	'z' encoded as 122
✓ '0' encoded as 48	'9' encoded as 57

Assigning values to char

```
char letter;    /* declare variable letter of type char */
```

```
letter = 'A';    /* OK */
```

```
letter = A;     /* NO! Compiler thinks A is a variable */
```

```
letter = "A";   /* NO! Compiler thinks "A" is a string */
```

```
letter = 65;   /* ok because characters are internally stored as numeric values (ASCII code) */
```

Floating-Point Types

- Floating-point types represent real numbers
 - Integer part
 - Fractional part
- The number 108.1517 breaks down into the following parts
 - 108 - integer part
 - 1517 - fractional part
- Floating-point constants can also be expressed in *scientific notation*. The value **1.7e4** represents the value **1.7×10^4** . The value before the letter **e** is known as the ***mantissa***, whereas the value that follows e is called the ***exponent***.
- There are three floating-point types
 - float
 - double
 - long double

SIZE AND RANGE OF VALUES FOR 16-BIT MACHINE FLOATING POINT TYPE

	Type	Size
Single Precision	Float	32 bits 4 bytes
Double Precision	double	64 bits 8 bytes
Long Double Precision	long double	80 bits 10 bytes

void

➤ 2 uses of void are

- To specify the return type of a function when it is not returning any value.
- To indicate an empty argument list to a function.

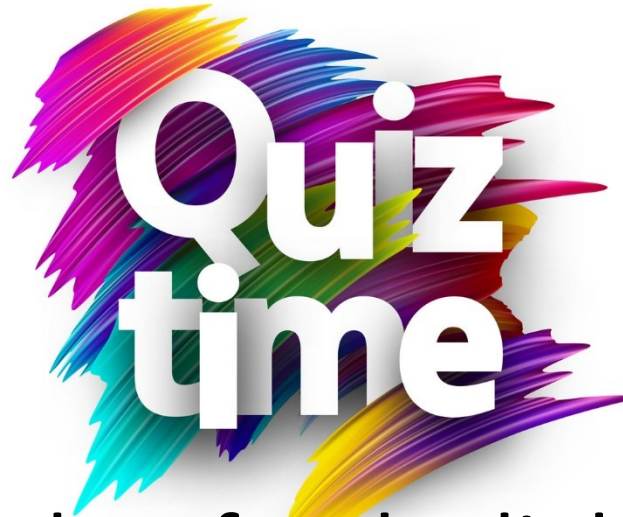
Best Practices for Programming

Naming Variables According to Standards

Prefix	Data Type	Example
✓ i	int and unsigned int	iTotalMarks
✓ f	float	fAverageMarks
✓ d	double	dSalary
✓ l	long and unsigned long	lFactorial
✓ c	signed char and unsigned char	cChoice
✓ ai	Array of integers	aiStudentId
✓ af	Array of float	afQuantity
✓ ad	Array of double	adAmount
✓ al	Array of long integers	alSample
✓ ac	Array of characters	acEmpName

Example: Using data types

```
#include <stdio.h>
int main ()
{
    int integerVar = 100;
    float floatingVar = 331.79;
    double doubleVar = 144368.4411;
    char charVar = 'W';
    printf("%d\n", integerVar);
    printf("%f\n", floatingVar);
    printf("%lf\n", doubleVar);
    printf("%c\n", charVar);
    return 0;
}
```



Go to posts/chat box for the link to the question

submit your solution in next 2 minutes

The session will resume in 3 minutes



Session 4 Summary

- Basic Data types: int, float, char, double and void
- ASCII format is used to encode character data (char)
- Floating point numbers (real numbers) can be stored in float, double or long double depending on the precision we want.