International Journal of Research in Engineering, Science and Management
Volume-2, Issue-12, December-2019
www.ijresm.com | ISSN (Online): 2581-5792

Analysis of Basic Data Types in C

R. Vishal[1], Vishal Vaman Mehtre[2]

[1]Student, Dept. of Electrical Engineering, Bharati Vidyapeeth Deemed University College of Engg., Pune, India
[2]Asst. Professor, Dept. of Electrical Engg., Bharati Vidyapeeth Deemed University College of Engg., Pune, India

***Abstract*: Increasing the demand of programing language so the data type is use which tells the processor what to expect and also the amount of spaced to be reserved by the variable. The data type is the type of data which are using for develop any program. The data type is an attribute of data which tells the compiler and interpreter how to use the data in programming language. The data types are expressed in syntax to declaration of memory location and variables. By the use of data types, it determines the type of operation for processing the data element.**

***Keywords*: C programming**

1. Introduction

A data type which specifies the range of value, it may be variable or constant. It also specifies how the value will be stored in a computer memory. It is similar to the data we use in our daily life. The 'c' language is very rich in its data types. The programmer selects the available variety of data types to the need of application as well as machine [1].

The data types are categories in four types

1. Primary data types
2. User defined data types
3. Derived data types
4. Empty data types

We use different types of data in our daily life. For example we use character data for writing name of any person, name of item, name of author of any book, for storing marks etc. similarly in any language different data types are available for handling different kinds of real life problems.

A. Primary data types

- Int
- Char
- Float
- Double

B. Integer data type

The data types which are use to store integer data is called integer data types. It consume 2 bytes of memory. It can not be store decimal numbers . integers can be signed or unsigned. The following figure shows how integer is stored in memory.

In printf and scanf %d is used as specifier for integer.

Example program on %d specifier

```
 main()
{
      int p, q;
      p = 6;
      q = 5;
      printf (“%d %d”, p, q);
}
```

First %d specify integer variable ‘p’.

Second %d specify integer variable ‘q’.

It can store range of value between – 32768 to +32767.

e.g. int p=33333 is not valid because 33333 > 32767.

C. Types of integers

1. Signed short integer
2. Short unsigned short integer
3. Signed long integer
4. Unsigned long integer

a) *Signed short integer (int):* It consume 2 bytes of memory. The format %d used as a specifier. it can store range between – 32768 to 32767.

b) *Unsigned short int:* It consume 2 bytes of memory. The format %u used as a specifier. It can store range of values between 0 to 65535.

c) *Signed long int:* It consume 4 bytes of memory. The format %ld is used as specifier. It can store ranges between -2147483648 to +2147483647.

d) *Unsigned long int:* It consume 4 bytes of memory. The format %lu is used as specifier. It can store the ranges between 0 to 4294967295.

D. Declaration of 4 types of integer data type

- Signed short integer
  Signed short int a
  /*declaration */
- Unsigned short integer
  Unsigned short int a;
  /*declaration */
- Signed long integer
  Long signed int a;
  /*declaration */
- Unsigned long integer
  Long unsigned int a;
  /*declaration */

E. Character data types

The data types which are used to store alphabet as well as all other 5chacters belongs to ASCII character set. It consumes 1 bytes of memory. If the single character is to be stored then %c is use as a specifier. If string (collection of character) is to be stored, then %s is used as a specifier. The following figure shows how the character is stored in memory.

Declaration of character data types is done by using ‘char’ keyword. The example of char data types when only one variable is to store.

```
main (
{
    Char a;
    a = ‘b’;
   printf (“%c”, a);
}
```

F. *Types of character data types*

1. Signed character data type
2. Unsigned character data type

Signed character and unsigned character both are consumed 1 byte of memory. For both format specifier is %c. The signed char can store range of values between -128 to +127 and the unsigned char can store valued between range 0 to 255[2]. Signed character is not use often.

G. *Float data types*

The float variable can store values with 6 to 7 digit of precision. It is also known as real data types. Float variable consume 4 bytes of memory. The format %f id used as a specifier for float variables. The ‘float’ keyword is used to declare float variable.

H. *Declaration of float variable*

```
main ()
{
   Float a, b, c;   /*declaration of float variable */
   a = 8;
   b = 4;
   c = a + b;
   printf (“%f”, c);
}
```

I. *Double data types*

The ‘double’ keyword is used to specify double data type. It can store the range of value with 14 to 15 digit of precision.

Types of double data type

- Double
- Long double
- *Double:* It consume 8 bytes of memory. The format %lf is used as a specifier. It can store the range of values between -1.7e308 to 1.7e308.

Example of double data type

```
Main ()
{
   double a, b, c;
   a = 5.673;
   b = 5.112;
   c = a + b;
   printf (“%lf”, c);
}
```

- *Long double:* It consume 10 bytes of memory. The format %ld is used as a specifier. It can store the range of values between 3.4e -4932 to 1.4e +4932.
-

Example of long double data type

```
# include<conio.h>
# include<stdio.h>
main ()
{
   Long double a, b, c;
   a = 5.673;
   b = 5.112;
   c = a + b;
   printf (“%ld”, c);
}
```

Derived data types

- Arrays
- Function
- Structure
- Pointer

J. *Arrays*

An arrays is defined as the collection of homogenous data element i.e. collection of related data items with similar data types and having a common name. The arrays of character is called string and the arrays of integer and float are A[1]………A[n-1].

Example:

Simply called array. The elements of array are represented by A[0], int Roll no [56]; means having Roll no array of 56 elements of integer data type.

K. *Function*

The function is defined as the block of statement which perform a particular task. The function facilitates division of large problem into smaller problems called subprogram. Each subprogram can be compiled and tested separately and can be reused in different program. In C++ there is a standard library which provides humorous inbuilt function that can be called in a program.

Example:

Memcpy ()   (copies one memory location to the another)

L. *Structure*

The structure is defined as the collection of heterogeneous data element i.e. collection of related items of different data types. The single structure may contain integer elements, character elements, floating point elements etc. These elements are called members.

Example

```
struct
      emp
    {
      char Name[20];
      int age;
      float salary;
  };
```

Name of structure is emp. It contains three members Name, Age, Salary.

M. Pointer

The pointer is defined as the variable that contains the address of another variablein memory. The variable is accessed by address operator ‘&’.

Example

```
int a =3
int*b =&a
int a = 3;
int*b = 3;
```

N. User defined data type

- Typedef
- Enum

O. Typedef

The explicitly new data type names can be allowed by using the keyword ‘typedef’. It defines the name for an existing type and it does not allow to create a new data class.

Example:

```
int main ()
  {
    BYTE a1, a2;
    a1 = ‘c’;
    cout << “  “  << a1;
    return 0;
  }
```

P. Enumerated

The enumerated data type is defines own data type and specify what values a variable of this type can take. The keyword ‘enum’ is used to define enumerated data type. It is used to assign the names to integral constants, the names make a program easy to read and maintain.

Example:

```
Enum week {
          Monday,
          Tuesday,
          Wednesday,
          Thursday,
          Friday,
          Saturday,
          Sunday
        };
int main ()
{
  enum week day ;
  day = Wednesday;
  cout << day;
  return 0;
}
```

Q. Empty data type

Void: The void is a data type which is used to specify, return type of function or simply it has no return type. Void means nothing.

Example:

Void display ()

- It indicates display function return nothing to function from where it is called. The keyword ‘void’ is used to specify void.

a) Characteristics of void data type

- Void consume no memory
- It also specifies that no argument is passd by function.

2. Conclusion

The analysis is carried out in ETAP software was useful in demonstrating the data types in ‘C’ programming language.

References

[1] Ishan Computer Programming & Applications

[2] Balaguruswamy, “Object Oriented Programming in C++”, Edition 2008, Tata McGraw Hill.

[3] Kanetkar Yashavant, "Letus C++", 2nd edition, BPB Publications.

[4] John Hubbard, “Schaum's Outline of Programming with C++,” 2nd edition, Schaum's series.

[5] Clifford A. Shaffer, “Data Structures and Algorithm Analysis in C++,” Third Edition, Dover Books on Computer Science.