CONSTANTS, VARIABLES & DATATYPES

BY
Dr. Md. Sherajul Islam

INRODUCTION

- A programming language is designed to help process certain kinds of data consisting of numbers, characters and strings to provide useful output known as information.
- The task of processing of data is accomplished by executing a sequence of precise instructions called program.
- These instructions are formed using certain symbols and words according to some rigid rules known as syntax rules.

CHARACTER SET:

- The characters in C are grouped into following categories:
  1) Letters 2) digits
  3) special characters 4) white spaces.

- Compiler ignores white spaces unless they are part of a string constant.

- White spaces may be used to separate words but prohibited between characters of keywords and identifiers.

- LETTERS: Uppercase A….Z, lower case a..z.

- DIGITS: All decimal digits 0..9

- SPECIAL CHARACTERS: comma(,), period(.), semicolon(;) , colon(:), question mark(?), quotation(“), dollar sign($), slash(/),back slash(\), percent sign(%), underscore(_), ampersand(&), asterisk(*), number sign(#).

- WHITE SPACES: Blank space, Horizontal tab, Carriage return, Newline, Form feed.

Trigraph characters:

- C introduces the concept of trigraph sequences to provide a way to enter certain characters that are not available on some keywords.

- Each trigraph sequence consists of three characters, 2 question marks followed by another character.

- Eg: ??= (number sign), ??)(right bracket]), ??((left bracket[) , ??! (vertical bar), ??< (left brace {) , ??> (right brace }),??/ (back slash).

C tokens:

- In a passage of text individual words and punctuation marks are called tokens.

- In a C program the smallest individual units known as C tokens.

- C has 6 types of tokens namely:

1. Keywords
   2) identifiers
   3)constants
   4)Strings
   5)special symbols
   6)operators.

Keywords and identifiers.

- Every C word is classified as either a keyword or an identifier.
- All keywords have fixed meanings and these meanings cannot be changed.
- Keywords serve as basic building blocks for program statements.
- All keywords must be written in lower case.
- The underscore character is also permitted in identifiers.
- It is usually used a link between two words in long identifiers

RULES FOR IDENTIFIERS:

1. First character must be an alphabet.
2. Must consist of only letters, digits or underscore.
3. Only first 31 characters are significant.
4. Cannot use keyword.
5. Must not contain white space.

CONSTANTS

- Constants refer to fixed values that do not change during the execution of program.

INTEGER CONSTANTS:

- An integer constant refer to a sequence of digits.

- There are 3 types of integers namely:
  Decimal integer, octal integer and hexadecimal integer.

- Decimal integer consist of a set of digits 0 through 9,precceded by an optional – or + sign.

- An octal integer constant consist of any combination of digits from the set 0 through 7. with a leading 0
- Eg: 037,0, 0456.

- A sequence of digits preceded by 0x or 0X is considered as hexadecimal integer.

- They may include alphabets A through F or f.

- Letter A through F represents numbesr 10 to 15.

REAL CONSTANTS:

- To represent quantities that vary continuously real constants are used.

- A real number may be expressed in exponential notation.
  SYNTAX: mantissa e exponent.

- Mantissa can be either real number expressed in decimal notation or an integer.

- Exponent is an integer number with an optional + or – sign.

- The letter ‘e’ separating the mantissa and the exponent, it can be written either lower case or upper case.
  SYNTAX: 0.65e4,12e-2.

- White space is not allowed.

- Exponential notation is useful for representing numbers that are either very large or very small in magnitude.

- Floating point constants are normally represented as double-precision quantities.

SINGLE CHARACTER CONSTANTS:

- A single character constant contains a single character enclose in a pair of single quote marks.

  Eg: ‘5’,’x’.

- Character constant ‘5’ is not same as number 5.

- Character constants have integer values known as ASCII values.

- Statement: printf (“%d”, ’a’); would print number 97, the ASCII value of letter ‘a’.

- Since each character constant represent an integer value, it is possible to perform arithmetic operations on character constants.

STRING CONSTANTS:

- A string constant is a sequence of characters enclosed in double quotes.

- Characters may be letters, numbers, special characters and blank spaces.

  Eg: “hello” , “1987”, “?...!”.

- Character constant is not equivalent to single character string constant.

BACK SLASH CHARACTER CONSTANTS:

- C supports some special back slash character constants that are used in output functions.

- These characters combinations are known as escape sequences.

- Back slash character constants are:
  ‘\a’ audible alert; ‘\b’ backspace; ‘\f’ form feed; ‘\n’ newline; ‘\r’ carriage return; ‘\t’ horizontal tab; ‘\v’ vertical tab; ‘\”single quote, ‘\?’ question mark; ‘\\’ backslash; ‘\0’ null.

VARIABLES

- A variable is a data name that may be used to store a data value.

- A variable may take different values at different times during execution.

- A variable can be chosen by the programmer in a meaningful way.

CONDITIONS FOR SPECIFYING VARIABLES:

1. They must begin with a letter. Some systems may permit underscore as first character.

1. Uppercase and lowercase are significant. The variable TOTAL is different from total and Total.

2. It should not be keyword.

3. Whitespace is not allowed.

4. Length should be normally more than 8 characters are treated as significant by many compilers.

- Examples of valid variables are:
  john, x1,T_raise, first_tag.

- Examples of invalid variables are:
  123,(area),25th,price$, %.

DATATYPES

- C language is rich in its data types.
- Storage representations and machine instructions to handle constants differ from machine to machine.
- The variety of datatypes allow to select the type appropriate to the needs of the application as well as machine.
- C supports 3 classes of datatypes:

  1)Primary datatypes
  2) derived datatypes
  3)derived datatypes.

- All C compilers support 5 fundamental datatypes namely: integer (int), character (char), floating point (float), double-precision floating point (double) and void.

- Many of them also offer extended datatypes such as long int, int ,long double.

Data type	Range of values
char	-128 to 127
int	-32768 to 32767
float	-3.4e+8 to 3.4e+8
double	1.7e-308 to 1.7e+308.

INTEGER TYPES:

- Integers are whole numbers with a range of values supported by particular machine.

- Integers occupy one word storage generally and since the word sizes of machine vary the size of integer that can be stored depends on computer.

- If we use 16-bit word length, the size of integer value is limited to range -32768 to 32767.

- If we use 32-bit word length can store an integer ranging from -2147483648 to 2147483647.

- In order to provide control over range of numbers and storage space C has 3 classes of integer storage namely: short int, int, long int in both signed and unsigned.

- Short int represents fairly small integer values and requires half amount as regular int number uses.

FLOATING POINT TYPE:

- Floating point numbers are stored in 32bits, with 6 digit precision.

- Floating point numbers are defined by keyword “float”.

- When accuracy is provided by a float number is not sufficient, double can be used to define number.
- A double datatype number uses 64 bits giving a precision of 14 digits.
- These are known as double precision number.
- Double datatype represent the same datatype that float represents but with greater precision.
- To extend the precision we may use long double which uses 80 bits.

VOID TYPES:

- Void type has no values. This is used to specify the type of functions.

- The type of function is said to be void when it doesn't return any value to the calling function.

CHARACTER TYPES:

- A single character can be defined as a character (char) type data.

- Characters are usually store in one byte of internal storage.

- Qualifier signed or unsigned may be used in char explicitly. Unsigned characters have values between 0 and 255, signed characters have values from -128 to 127

Data Type	Maximum	Minimum	Bytes
int	32767	-32768	2
unsigned int	65535	0	2
short	32767	-32768	2
long	2147483647	-2147483648	4
char (ASCII codes)	127	-128	1
unsigned char	255	0	1
float	3.4E+38	3.4E-38	4
double	1.7E+308	1.7E-308	8

```
const float PI = 3.1415;
```