VARIABLES IN C PROGRAMMING

Computers are useful because of their ability to store and manipulate huge amounts of information. This information may be numbers, such as a financial report, or alphabetic characters like names and addresses. Managing this information requires the usage of programming languages. One of the important task of a programming language is identifying the type of data it is manipulating.

Data is stored in a computer's memory. The memory system comprises of uniquely numbered cells called memory addresses. We need to know the address where something is stored in order to retrieve it and work on it. A programming language frees us from keeping track of these memory addresses by substituting names for them. These names are called variables. Variables are descriptive names for the memory addresses.

Before we use a variable in C we must declare it. We must identify what kind of information will be stored in it. This is called defining a variable.Variables must be declared at the start of any block of code, but most are found at the start of each function. A variable must be defined to be one of the legal C data types. When a variable is defined it is not automatically initialized, it is the responsibility of the programmer to initialize this to a start value.

Variable names

All variable definitions must include two things variable name and its data type. Some rules to be followed in naming a variable in C are, it must start with an alphabet and can contain letter, underscores and digits. Both uppercase and lowercase letters can be used. Spaces should not be used in a variable name. Certain keywords like *int*, *float*, *struct*, *if*, *while* cannot be used as variable names. The variable names should not be very long and one should refer to the documentation of the C compiler to know the limitation. Generally the first eight characters of a variable name are significant.

Some of the valid variable names are :

fAmount
iTotal
iCounter

Note : The variable names should be meaningful and denote what the variable stores. As far as possible single letter variable names should be avoided.

C provides a number of data types, some of the basic types used are :

int	integer
char	character

float	a floating point number

Table 3.1

There are variants to the above, such as

unsigned int	unsigned integer
short int	short integer
long int	long integer
double	double precision floating point number

Table 3.2

When declaring variables to be of type long int, short int and unsigned int it is permissible to omit the keyword int. Since these variables must always be integers. Thus we can declare an unsigned integer counter as

unsigned counter;

Unlike in programming languages like Visual Basic, there is no datatype such as string in C. However you can declare an array of the datatype char and write functions to manipulate this.

The main difference among these data types is the amount of memory allocated for storing these. The maximum and minimum values that can be stored in these variables depends on the version of C and the computer system being used. Some of the typical values are :

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

Table 3.3

Format for declaring a variable in C is :

var_type variable names;

Eg :

int counter;

float total, amount, interest;

char answer;

There is no Boolean type in C, you should use int, unsigned char or char type.

Type int

Int is an integer datatype, a variable declared of this type can have any value in the range indicated in Table 3.1 above. This is used to store whole numbers.

Program 3.1

```
/* Program to declare an integer variable i and then set it to be 20
*/
#include <stdio.h>
void main()
  {
int i = 20;
printf("This is my integer: %d \n", i);
  }
```

The %d is to tell printf() routine to format the integer i as a decimal number.

Type float

Floating point variables can store a value containing a decimal point. Floating point numbers may contain both a whole and fractional part.

Program 3.2

```
/* Program to declare a float variable f and then set it to be 3.1415
*/
#include <stdio.h>
void main()
  {
float f = 3.1415;
printf("This is my float: %f \n", f);
  }
```

The %f is to tell printf() routine to format the float variable f as a decimal floating number. Now if you want to see only the first 2 numbers after the floating point, you would modify your printf() routine call to be like this:

```
printf("This is my shorter float: %.2f \n", f);
```

Type double

The type double is similar to float except that it is used whenever the accuracy of a floating point variable is insufficient. Variables declared to be of type double can store approximately twice as many significant digits as a variable of type float. You can use similar method as in the previous example to print a double.

Type char

Character variables are used to store character values particularly single characters.

Note: The keyword *void* is used to denote nothing. As such one will not declare a variable of type void in a program, but it is more useful when declaring functions which do not return a value.

Program 3.3

```
/* Program to declare a character variable d and then set it to be
'd' */
#include <stdio.h>
void main()
  {
char c;
c = 'd';
printf("This is my character: %c \n", c);
  }
```

The %c is to tell printf() routine to format the variable f as a character.

Source : http://www.peoi.org/Courses/Coursesen/cprog/frame3.html