

C Program

Professor Dr. M. Ismail Jabiullah

Professor

Department of CSE

Daffodil International University

Bangladesh

C Program

Basic Program Structures and Variables

- C Program Structure
- Data Types
- Concept of Variables
- Variable Naming Conventions
- C Reserve Words

Introduction

- The C programming language was designed by **Dennis Ritchie** at **Bell Laboratories** in the early 1970s

Influenced by

- ALGOL 60 (1960),
 - CPL (Cambridge, 1963),
 - BCPL (Martin Richard, 1967),
 - B (Ken Thompson, 1970)
-
- Traditionally used for systems programming, though this may be changing in favor of C++
-
- Traditional C:
 - *The C Programming Language*, by Brian Kernighan and Dennis Ritchie, 2nd Edition, Prentice Hall
 - Referred to as *K&R*

Standard C

- Standardized in 1989 by **ANSI** (American National Standards Institute) known as **ANSI C**
- International standard (ISO) in 1990 which was adopted by **ANSI** and is known as **C89**
- As part of the normal evolution process the standard was updated in 1995 (**C95**) and 1999 (**C99**)
- **C++ and C**
 - C++ extends C to include support for Object Oriented Programming and other features that facilitate large software development projects
 - C is not strictly a subset of C++, but it is possible to write “*Clean C*” that conforms to both the C++ and C standards.

Elements of a C Program

- A C development environment includes
 - **System libraries and headers:** a set of standard libraries and their header files. For example see /usr/include and glibc.
 - **Application Source:** application source and header files
 - **Compiler:** converts source to object code for a specific platform
 - **Linker:** resolves external references and produces the executable module
- User program structure
 - There must be one main function where execution begins when the program is run. This function is called main
 - `int main (void) { ... },`
 - `int main (int argc, char *argv[]) { ... }`
 - `int main (int argc, char *argv[], char *envp[])`
 - additional local and external functions and variables

C Program Structure

- C programs are composed of one or more functions and at least one function must have name main().
- If there is only one function and that should be main().
- C program start its execution from the function main().
- Function must have
 - Function heading with () symbol and all arguments are in the brackets ()
 - Arguments must have declared before use
 - Function have all of its actions in the function body.

```
/*program to calculate the sum of two integer numbers*/  
#include <stdio.h>  
void main()  
{  
    int a, b, s;  
    printf("\nEnter First Number: ");  
    scanf("%d",&a);  
    printf("\nEnter Second Number: ");  
    scanf("%d",&b);  
    s=a+b;  
    printf("\nSum = %d",s);  
}
```

Annotations for the code above:

- Comment
- Header file preprocess
- Function heading
- Function start
- Variable declaration
- Message display
- variable input
- Value of expression assign
- Output of the value of variable
- Function end

Characters in C

C programs consist of

- Constants
- Variables
- Operators
- Expressions

All C fundamental components are written as by

- **English Capital letters:** A to Z
- **English small letters:** a to z
- **Numerals :** 0 to 9
- **And special characters in the keyboard:** !, *, @, #, \$, %, &, (,), {, }, \, /, ?, <, >, =, etc.

C Reserve Words

C programs have some reserved words or keywords and they are:

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

Identifiers

Identifier

- Identifiers are the names of variables, function names, array names, program names etc.
- **Rules of naming Identifiers**
 - English characters a to z, 0 to 9, and underscore (_) sign
 - First letter must be letter a to z
 - No blank space should be allowed within the name
 - Maximum 31 characters are allowed.

- **Examples:**

- a7, sum, average, product, quotient,
increment, for9

are valid identifiers.

- 2sum, totalsum's, sum-1, average 8,
product?, const, switch

are not valid identifiers.

Identifiers (Exercise)

Identify the Valid Identifiers (with Explanation)

1. Aaaaaa7
2. add7sum
3. findaverage
4. 9product
5. 109quotient
6. increment_decrement
7. for999
8. valid-identifiers
9. 2*sum
10. Total=sum's
11. sum-165
12. average 8
13. product?7
14. Const
15. notvalididentifiers

Data Types in C

C has several data types and they are:

- Integer
- Character
- Floating-point numbers
- Double-precision Numbers

All data types have four modifiers and they are:

- signed
- unsigned
- long
- short

Data Ranges of Each Data Types in C

Data Type	Data Type Description	Memory Size
char	Character Types	1 byte
int	Integer Type Numbers	2 bytes or 1 word
float	Floating-point Numbers	4 bytes or 2 words
double	Double-Precision Numbers	8 bytes or 4 words

Data Type	Size of Memory	Memory Range
char	1 byte	-128 to +127
signed char	1 byte	-128 to +127
unsigned char	1 byte	0 to 255
int	2 bytes	-32768 to +32767
signed int	2 bytes	-32768 to +32767
unsigned int	2 bytes	0 to 65535
short in	2 bytes	-32768 to +32767
long int	4 bytes	-2147483648 to +2147483647
unsigned long int	4 bytes	0 to 4294967295

Constants

Constants in C are 4 kinds:

- **Integer Constants**
- Character Constants
- Floating-point Constants
- String Constants
- **Integer Constants**
 - No comma, no blank space are in the numbers
 - Only negative sign (-) can be place in front of the constant
 - Value must be within the defined upper and lower value
 - **Examples:** 0 1 743 54456 876549
are valid integer constants.
 - 01 12 3 +345 99999999
are not valid integer constants

Integer Constants (Exercise)

1. 01987
2. 1,7433
3. 544516
4. 876549
5. 01111
6. 12 35678
7. +1111345
8. 99999999
9. 10,000,000
10. -19098
11. 1908887
12. 3290000.00

Character Constants

Constants in C are 4 kinds:

- Integer Constants
- **Character Constants**
- Floating-point Constants
- String Constants
- **Character Constants**
 - A single character or symbol within a single quotation marks
 - **Examples:** 'a' '1' '&' '@' ' ' are valid integer constants.
 - a' '11' '&6' '@' " " are not valid integer constants.

Character Constants (Exercise)

1. 'ab'
2. '7'
3. '&'
4. '%'
5. '“
6. 'a'
7. '1981'
8. '&6'
9. '@'
10. “ “
11. 'w'
12. '0'
13. '-'
14. "0"

Floating-point Constants

Constants in C are 4 kinds:

- Integer Constants
- Character Constants
- **Floating-point Constants**
- String Constants
- **Floating-point Constants**
 - All number with decimal point (.) are floating-point numbers
 - No comma, no blank space are in the numbers
 - Only negative sign (-) can be place in front of the constant
 - Value must be within the defined upper and lower value
 - **Examples:** 0. 0.8 873.22 2E-6 0.6E-3 are valid floating-point numbers
 - 0 .8 33556677873.22 2E-16 0.6E-0.3 are not valid floating-point numbers

Floating-point Constants (Exercise)

1. 19800.
2. 23450.8
3. 12873.220000
4. 209E-6
5. 120.6E-3
6. 0
7. .8
8. 12333556677873.22
9. 2E-16
10. 0.6E-0.3
11. E+123
12. 99999.9999
13. 018978.20
14. 20.018978

String Constants

Constants in C are 4 kinds:

- Integer Constants
- Character Constants
- Floating-point Constants
- **String Constants**

- **String Constants**

- Texts or numbers written within the double quotation sign (“”) pair.

- **Examples:** “jamal” “yellow” “ ” “123”

are valid string constants

- `jamal` yellow” “Jamal ‘123”

are not valid string constants

String Constants (Exercise)

1. "jamal"
2. "redyellow"
3. "-----"
4. "1234567890"
5. `jamal`
6. Yellow"
7. "Jamal
8. '123"
9. "Jamal Uddin Russel of CSE Department"
10. 'Daffodil International University'
11. "Daffodil; International; University"

Escape Sequences

A character written in within double quotation mark (“”) and using a slash sign (\) is called escape sequences.

Escape Sequence	Equal ASCII Value	Functions
\a	007	Ring a alert bell
\b	008	Back Space
\t	009	Tab key
\n	010	New line
\”	034	Double Quotation mark
\’	039	Single Quotation Mark
\\	092	Back slash mark
\?	063	Quotient Mark

C Variables

- Variable is a name that contains the value of the data item.
- Variables contains the value as of its data types.
- Variables must be declared before of its use.

Rules of Variable Names:

- English characters a to z, 0 to 9, and underscore (_) sign
- First letter must be letter a to z
- No blank space should be allowed within the name
- Maximum 31 characters are allowed.
- **Examples:**
 - a7, sum, average, product, quotient,
increment, for9 **are valid variable names.**
 - 2sum, totalsum's, sum-1, average 8,
product?, const, switch **are not valid variables.**

Examples:

```
char z, w;  
float i, j, l;  
int x, y, z, w;
```

C Variables (Exercise)

1. A7
2. Sum
3. Average
4. Product
5. Quotient
6. increment
7. 2sum
8. totalsum's
9. sum-1
10. average 8
11. Product?
12. Const
13. char
14. Z
15. W\
16. Float
17. j
18. L
19. x
20. y
21. z
22. w

C Expressions

C expressions are of the following types:

Expression or assignment statement:

```
a = 32;
```

```
c = a+b;
```

```
++i;           or           i=i+1;
```

Compound Statement:

```
{  
    int a,b,x;  
    a=32;  
    b=43;  
    x=a+b;  
    printf("\nSum = %d", x);  
}
```


Some C Programs

1. C Programs

Problem: A C Program to display the name address with a message.

```
/*program 01: to display the name and address*/
#include<stdio.h>
int main()
{
    printf("Jamal Uddin Russel\n");
    printf("Department of Computer Science and Engineering\n");
    printf("Daffodil International University\n");
    printf("Bangladesh\n");
    printf("Programming is fun\n");
    printf("Welcome to Programming World\n");
    return 0;
}
```

2. C Programs

Problem: A C program that finds the sum, subtract and product of two given numbers.

```
/*program 02: to find the sum, subtract, product of given two numbers*/
#include<stdio.h>
int main()
{
    int i,j,sum,subtract,product;
    i=250;
    j=100;
    sum=i+j;
    subtract=i-j;
    product=i*j;
    printf("\n\nSum = %d",sum);
    printf("\n\nSubtract = %d",subtract);
    printf("\n\nProduct = %d",product);
    return 0;
}
```

3. C Programs

Problem: A C program that enters two numbers from the keyboard and finds the sum, subtract and product of them

```
/*program 03: to find the sum, subtract, product of two numbers where numbers are inputted from the keyboard*/
```

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int i,j,sum,subtract,product;
```

```
    printf("\nEnter first number: ");
```

```
    scanf("%d",&i);
```

```
    printf("\nEnter second number: ");
```

```
    scanf("%d",&j);
```

```
    sum=i+j;
```

```
    subtract=i-j;
```

```
    product=i*j;
```

```
    printf("\n\nSum = %d",sum);
```

```
    printf("\n\nSubtract = %d",subtract);
```

```
    printf("\n\nProduct = %d",product);
```

```
    return 0;
```

```
}
```

4. C Programs

Problem: A C program that enters two numbers from the keyboard and finds the sum, subtract and product of them with formatted output.

/*program 04: to find the sum, subtract, product of given two numbers where numbers are inputted from the keyboard*/

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int i,j,sum,subtract,product;
```

```
    printf("\nEnter first number: ");
```

```
    scanf("%d",&i);
```

```
    printf("\nEnter second number: ");
```

```
    scanf("%d",&j);
```

```
    sum=i+j;
```

```
    subtract=i-j;
```

```
    product=i*j;
```

```
    printf("\n\nSum of %d and %d is = %d",i,j,sum);
```

```
    printf("\n\nSubtract of %d and %d is = %d",i,j,subtract);
```

```
    printf("\n\nProduct of %d and %d is = %d",i,j,product);
```

```
    return 0;
```

```
}
```

More Programs

Program 1: A C program that display the following message

**Programming is fun
C programming is also very fun
Welcome to the World of C**

```
/*program 1 to display the following message:  
Programming is fun  
C programming is also very fun  
Welcome to the World of C */  
#include<stdio.h>  
void main()  
{  
    printf("Programming is fun");  
    printf("C Programming is also very fun");  
    printf("Welcome to the world of C");  
}
```

Program 2: A C program that display the following message as formatted

**Programming is fun
C programming is also very fun
Welcome to the World of C**

/*program 2 to display the following message as formatted

```
Programming is fun  
C programming is also very fun  
Welcome to the World of C */  
#include<stdio.h>  
void main()  
{  
    printf("\nProgramming is fun");  
    printf("\nC Programming is also very fun");  
    printf("\nWelcome to the world of C");  
}
```


Program 03: A C program that finds the sum of two given integer values.

```
/*program 03 to display the sum*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s;  
    a=120, b=210;  
    printf("Sum = %d",s);  
}
```

Program 04 : A C program that display the sum of two given integer values with the given values

```
/*program 4 to display the sum and the numbers*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s;  
    a=120, b=210;  
    printf("a = %d",a);  
    printf("b = %d",b);  
    printf("Sum = %d",s);  
}
```

Program 5: A C program to display the sum and the numbers with proper format

```
/*program 5 to display the sum and the numbers with proper format*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s;  
    a=120, b=210;  
    printf("\na = %d",a);  
    printf("\nb = %d",b);  
    printf("\nSum = %d",s);  
}
```

Program 6: A C program to display the sum of two numbers with the two numbers as specified format

```
/*program 6 to display the sum of two numbers with the two numbers as specified  
format*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s;  
    a=120, b=210;  
    s=a+b;  
    printf("\nFirst Number = %d",a);  
    printf("\nSecond Number = %d",b);  
    printf("\nSum = %d",s);  
}
```

Program 7: A C Program that display the sum, subtract and product with the given numbers and specified format

```
/*program 7 to display the sum, subtract, product with the format*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s,sb,pr;  
    a=120, b=210;  
    s=a+b;  
    sb=a-b;  
    pr=a*b;  
    printf("\nFirst Number = %d",a);  
    printf("\nSecond Number = %d",b);  
    printf("\nSum = %d",s);  
    printf("\nSubtract = %d",sb);  
    printf("\nProduct = %d",pr);  
}
```

Program 8: A C Program that display the sum, subtract and product with the numbers taken from the keyboard and specified format

```
/*program 8 to display the sum, subtract, product of the two numbers taken from  
the keyboard with the format*/  
#include<stdio.h>  
void main()  
{  
    int a,b,s,sb,pr;  
    printf("\nEnter First Number = %d",a);  
    scanf("%d",&a);  
    printf("\nSecond Number = %d",b);  
    scanf("%d",&b);  
    s=a+b;  
    sb=a-b;  
    pr=a*b;  
    printf("\nFirst Number = %d",a);  
    printf("\nSecond Number = %d",b);  
    printf("\nSum = %d",s);  
    printf("\nSubtract = %d",sb);  
    printf("\nProduct = %d",pr);  
}
```

Thanks