

NAME: Pushkar Kanjani

**ROLL NUMBER: 23TBIT177**

**NEW ROLL NUMBER: 23BIT153**

**ICT DIVISION-2 H4**

**SUBJECT: C Programming Lablist**

## **History about C-Programming language:**

The C programming language, developed by Dennis Ritchie at Bell Labs in the early 1970s, is a cornerstone of modern computing. Born from the desire to enhance the B programming language, C quickly evolved into a powerful and flexible tool. Its concise syntax and low-level memory access made it suitable for system programming, which was crucial for developing operating systems like Unix.

Dennis Ritchie's work on Unix played a pivotal role in C's popularity. He rewrote Unix in C, showcasing the language's portability and efficiency. The publication of "The C Programming Language" by Brian Kernighan and Dennis Ritchie in 1978 further solidified its status. C's influence extended beyond systems programming; it became a foundation for software development and inspired languages like C++, C#, and Objective-C.

The standardization of C by ANSI in 1989 and ISO in 1990 ensured its wide adoption and cross-platform compatibility. Its impact endures in modern programming languages and systems, attesting to its enduring significance in shaping the digital world.

# UNIT 1

## C basics on Input, Output constructs and different data types

### Theory

```
#include ----- > Header  
  
int main (void) ----- > Main  
{ printf ("Hello World") -----> Statement  
return 0 ; ----- > Return }
```

#### 1.Header:

All lines that start with # are processed by a preprocessor which is a program invoked by the compiler. In the above example, the preprocessor copies the preprocessed code of stdio.h to our file. The .h files are called header files in C.

#### 2. Main Method Declaration - [int main()]

The next part of a C program is to declare the main() function. It is the entry point of a C program and the execution typically begins with the first line of the main(). The empty brackets indicate that the main doesn't take any parameter. The int that was written before the main indicates the return type of main(). The value returned by the main indicates the status of program termination.

#### 3. Body of Main Method – [enclosed in {}]

The body of a function in the C program refers to statements that are a part of that function. It can be anything like manipulations, searching, sorting, printing, etc. A pair of curly brackets define the body of a function. All functions must start and end with curly brackets.

#### **4. Statement – [printf(“Hello World”);]**

Statements are the instructions given to the compiler. In C, a statement is always terminated by a semicolon (;). In this particular case, we use printf() function to instruct the compiler to display “Hello World” text on the screen.

#### **5. Return Statement – [return 0;]**

The last part of any C function is the return statement. The return statement refers to the return values from a function. This return statement and return value depend upon the return type of the function. The return statement in our program returns the value from main(). The returned value may be used by an operating system to know the termination status of your program. The value 0 typically means successful termination.

#### **Printf:**

printf function is used to print output on the screen. This function is a part of the C standard library “stdio.h” and it can allow formatting the output in numerous ways.

#### **Scanf:**

scanf is a function that stands for Scan Formatted String. It is used to read data from stdin (standard input stream i.e. usually keyboard) and then writes the result into the given arguments.

- It accepts character, string, and numeric data from the user using standard input.
- scanf also uses format specifiers like printf.

### **There are some common data types in C –**

- int – Used to store an integer value.
- char – Used to store a single character.
- float – Used to store decimal numbers with single precision.
- double – Used to store decimal numbers with double precision.

### **Hardware Used:**

Processor: 12th Gen Intel(R) Core(TM) i5-12500H 2.50 GHz

Installed RAM: 16.0 GB (15.6 GB usable)

System type 64-bit operating system, x64-based processor

Edition: Windows 11 Home Single Language

Core:12 Logical Processor:16

### **Software Used:**

Text editor: Visual Studio Code

Compiler: GCC(By MinGw)

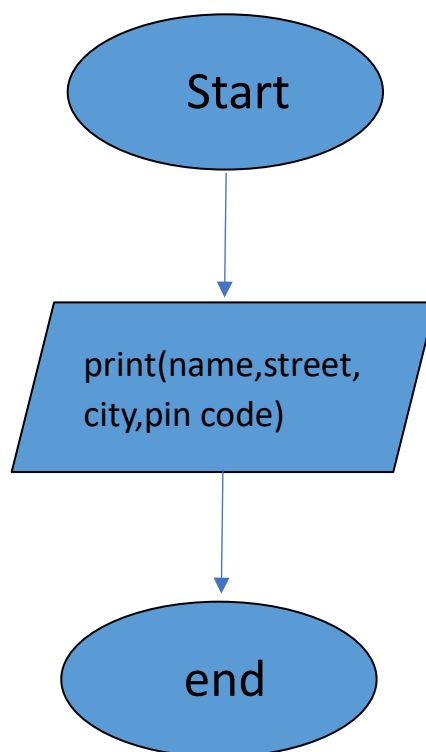
**Q1 (a) Write a program that will print your mailing address in the following form:**

**First line : Your Name**

**Second line : Door No, Street**

**Third line : City, Pin code**

**FLOWCHART:**



## SOURCE CODE:

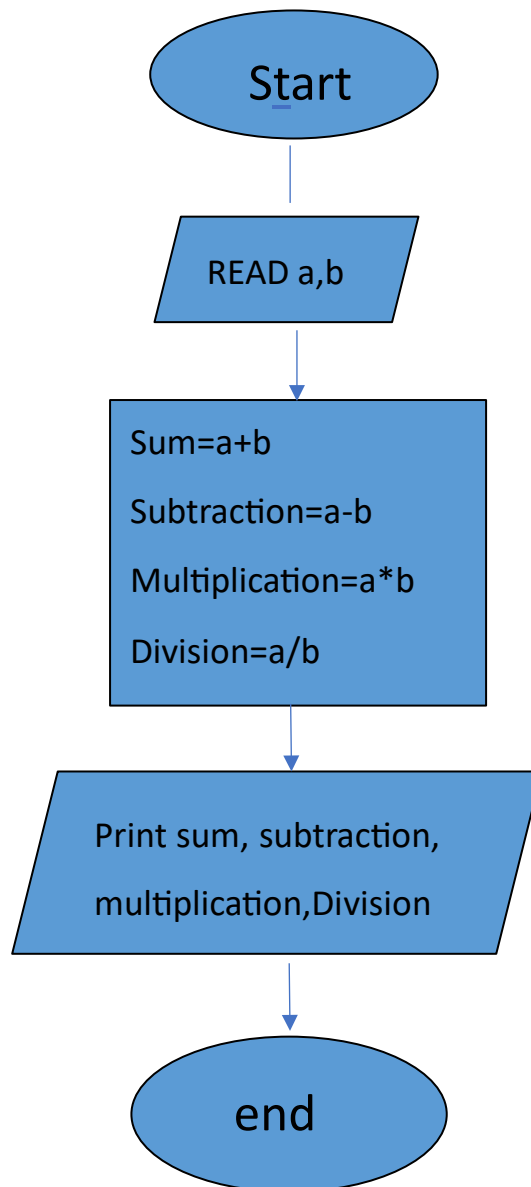
```
#include<stdio.h>
int main()
{
    printf("NAME: PUSHKAR KANJANI \nROLL NUMBER: 23TBIT177");
    printf("\nPushkar Kanjani \n105, PDEU Road \nGandhinagar,
328007");
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
Pushkar Kanjani
105, PDEU Road
Gandhinagar, 328007
```

**(b) Write a program add, subtract, multiply, divide two numbers by asking inputs from the user.**

FLOWCHART:





## SOURCE CODE:

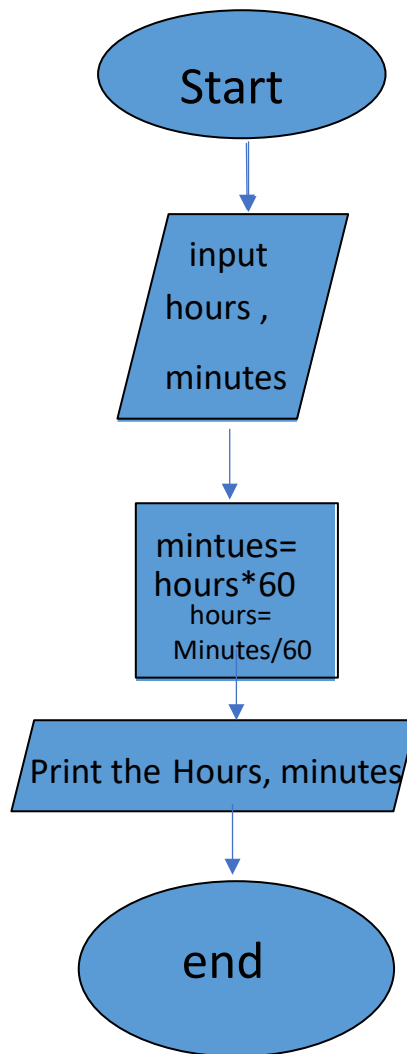
```
# include<stdio.h>
int main()
{
    int a,s,m,d,b,c;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("ENTER TWO VALUES");
    scanf("%d", &b);
    scanf("%d", &c);
    a=b+c;
    printf("%d is sum of %d and %d",a,b,c);
    s=b-c;
    printf("\n%d is subtraction of %d and %d",s,b,c);
    m=b*c;
    printf("\n%d is multiple of %d and %d",m,b,c);
    d=b/c;
    printf("\n%d is dividend of %d and %d",d,b,c);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER TWO VALUES 10 5
15 is sum of 10 and 5
5 is subtraction of 10 and 5
50 is multiple of 10 and 5
2 is dividend of 10 and 5
```

(c) Write a program to convert hours into minutes and minutes to hours.

FLOWCHART:



SOURCE CODE:

```
#include<stdio.h>
int main()
{
    int a,b,c,d;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUES IN HOURS");
    scanf("%d", &a);
```

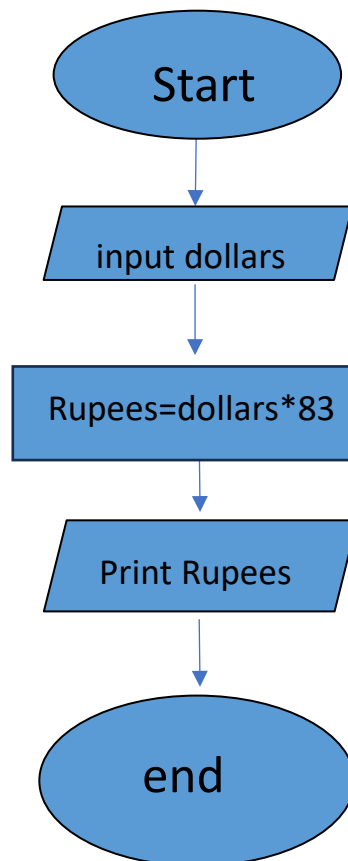
```
c=a*60;
printf("%d minutes is equal to %d hours",c,a);
printf("\nENTER VALUES IN MINUTES");
scanf("%d", &b);
d=b/60;
printf("%d hours is equal to %d minutes",d,b);
return 0;
}
```

OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUES IN HOURS
5
300 minutes is equal to 5 hours
ENTER VALUES IN MINUTES
360
6 hours is equal to 360 minutes
```

**(d) Write a program to convert dollars into rupees from the user's input.**

FLOWCHART:



## SOURCE CODE:

```
#include<stdio.h>
int main()
{
    float a,b ;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUE IN DOLLARS");
    scanf("%f", &a);
    b=a*83;
    printf("%.2f rupees is equal to %.2f dollars",b,a);
    return 0;
}
```

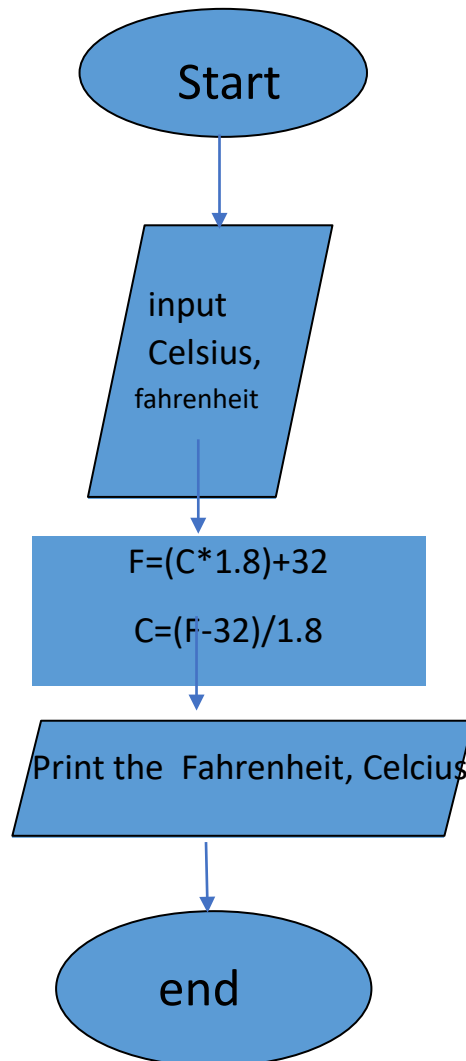
## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUE IN DOLLARS
67.7
5619.10 rupees is equal to 67.70 dollars
```

(e) Write a program to convert Celsius to Fahrenheit and Fahrenheit to Celsius

$$C = (F - 32) / 1.8$$

FLOWCHART:



## SOURCE CODE:

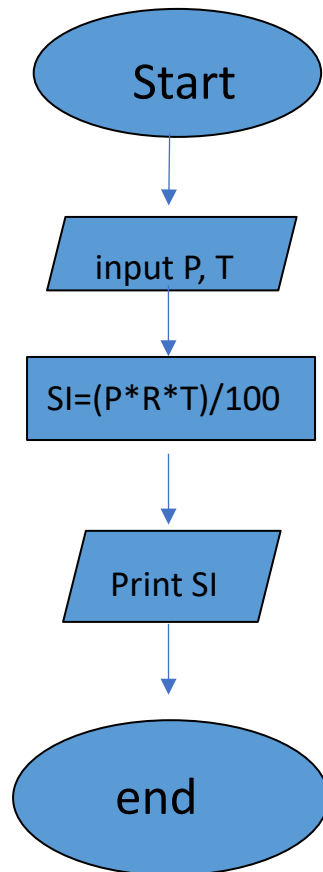
```
#include<stdio.h>
int main()
{
    float a,b,c;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUE IN FAHRENHEIT");
    scanf("%f", &a);
    b=(a-32)/1.8;
    printf("%.2f Celcius is equal to %.2f Fahrenheit",b,a);
    printf("\nENTER VALUE IN CELCIUS");
    scanf("%f", &a);
    c=(a*1.8)+32;
    printf("%.2f Fahrenheit is equal to %.2f Celcius",c,a);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUE IN FAHRENHEIT
145
62.78 Celcius is equal to 145.00 Fahrenheit
ENTER VALUE IN CELCIUS
1000
1832.00 Fahrenheit is equal to 1000.00 Celcius
```

**(f) Write a program to find the Simple Interest from the user principal and time with fixed rate of interest as 10%.**

FLOWCHART:





## SOURCE CODE:

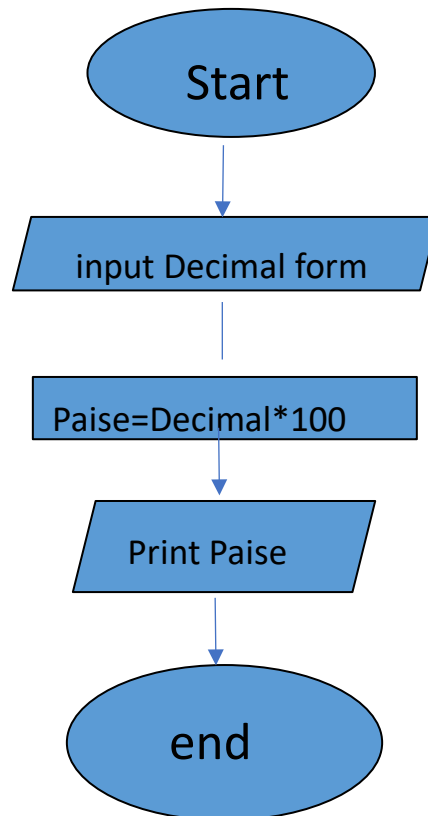
```
#include<stdio.h>
int main()
{
    float p,r,t,i;
    r=10;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUES OF PRINCIPLE AMOUNT AND TIME PERIOD TO
CALCULATE SIMPLE INTREST");
    scanf("%f", &p);
    scanf("%f", &t);
    i=p*r*t/100;
    printf("%f is your simple intrest", i);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUES OF PRINCIPLE AMOUNT AND TIME PERIOD TO CALCULATE SIMPLE INTREST
3300
7
2310.00 is your simple intrest
```

**(g) Write a program to read the price of an item in decimal form (like 15.95) and print the output in paise (like 1595 paise).**

**FLOWCHART:**



## SOURCE CODE:

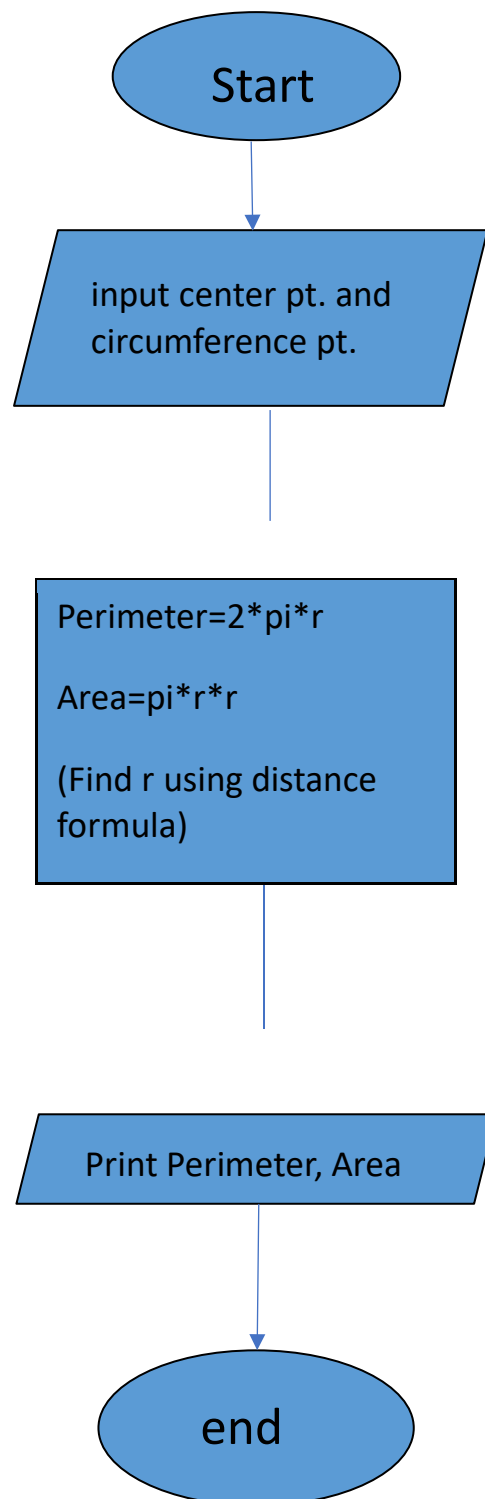
```
#include<stdio.h>
int main()
{
    float a,b ;
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUE IN RUPEES");
    scanf("%f", &a);
    b=a*100;
    printf("%.2paise is equal to %.2f rupees",b,a);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUE IN RUPEES
15.95
1595.00 paise is equal to 15.95 rupees
```

(h) A point on the circumference of a circle whose centre is (o, o) is (4,5). Write a program to compute the perimeter and area of the circle.

FLOWCHART:



## SOURCE CODE:

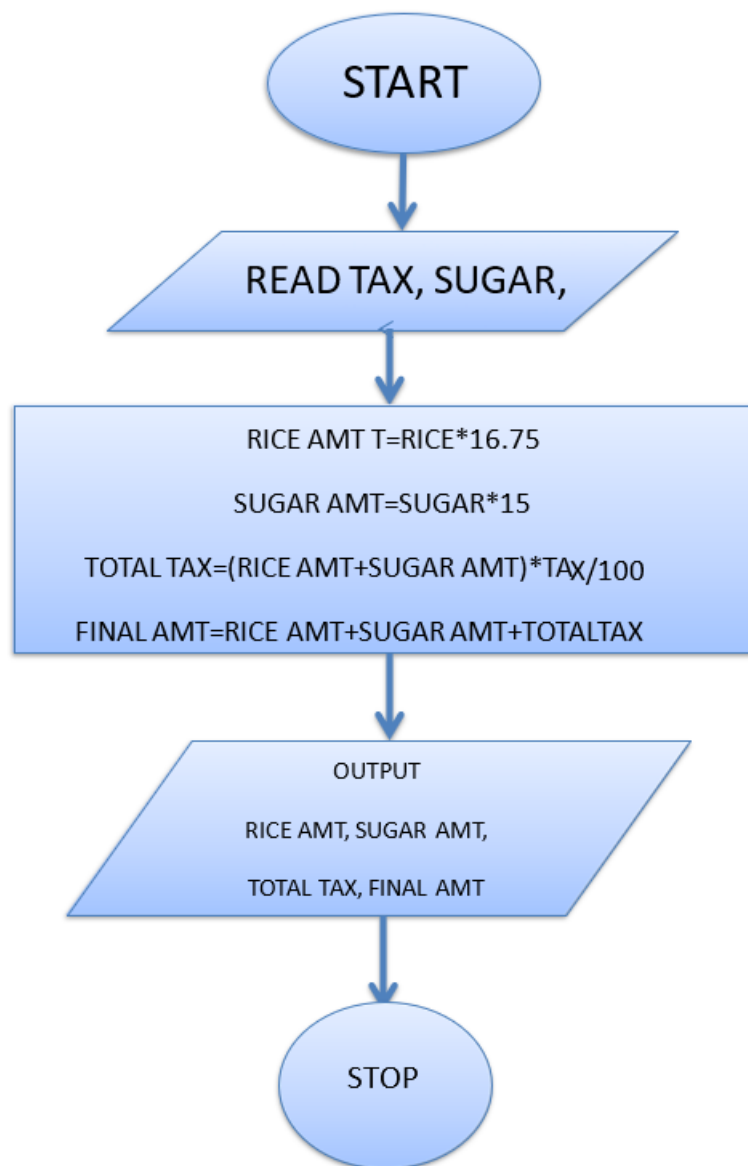
```
#include<stdio.h>
#include<math.h>
Int main()
{
    float l,b,r,p,a;
    l=4;
    b=5;
    r=sqrt(l*l+b*b);
    printf("NAME: PUSHKAR KANJANNI \nROLL NUMBER: 23TBIT177");
    p=2*3.14*r;
    a=3.14*r*r;
    printf("\nwhere %0.2f and %0.2f are perimetr and area
respectively",p,a);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
where 40.21 and 128.74 are perimetr and area respectively
```

- (i) The price of one kg of rice is Rs. 16.75 and one kg of sugar is Rs. 15. Write a program to get the quantity to be bought from the user and display the total bill in a formatted manner as we get from the supermarket.

FLOWCHART:



## SOURCE CODE:

```
#include<stdio.h>
int main()
{
    float a,b,c,d,e,f;
    e=16.75, f=15;
    printf("NAME: PUSHKAR KANJANI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER QUANTITY OF RICE AND SUGAR IN KG");
    scanf("%f",&a);
    scanf("%f",&b);
    c=e*a;
    d=f*b;
    printf("%f RUPEES OF THE RICE BOUGHT \n%fRUPEES OF SUGAR
BOUGHT",c,d);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER QUANTITY OF RICE AND SUGAR IN KG
3 5
50.25 RUPEES OF THE RICE BOUGHT
75.00RUPEES OF SUGAR BOUGHT
```

## **UNIT 2**

### **Types of Operators, Precedence and associativity of Operators**

**Some of the commonly used operators in the C programming language along with their definitions:**

#### **Arithmetic Operators:**

+ (Addition): Adds two operands.

- (Subtraction): Subtracts the right operand from the left operand.

\* (Multiplication): Multiplies two operands.

/ (Division): Divides the left operand by the right operand.

% (Modulus): Returns the remainder when the left operand is divided by the right operand.

#### **Relational Operators:**

== (Equal to): Checks if two operands are equal.

!= (Not equal to): Checks if two operands are not equal.

< (Less than): Checks if the left operand is less than the right operand.

> (Greater than): Checks if the left operand is greater than the right operand.

<= (Less than or equal to): Checks if the left operand is less than or equal to the right operand.



`>=` (Greater than or equal to): Checks if the left operand is greater than or equal to the right operand.

### **Logical Operators:**

`&&` (Logical AND): Returns true if both operands are true.

`||` (Logical OR): Returns true if at least one operand is true.

`!` (Logical NOT): Returns the opposite of the operand's truth value.

### **Increment and Decrement Operators:**

`++` (Increment): Increases the value of the operand by 1.

`--` (Decrement): Decreases the value of the operand by 1.

### **Bitwise Operators:**

`&` (Bitwise AND): Performs a bitwise AND operation.

`|` (Bitwise OR): Performs a bitwise OR operation.

`^` (Bitwise XOR): Performs a bitwise exclusive OR operation.

`~` (Bitwise NOT): Flips the bits of the operand.

`<<` (Left shift): Shifts the bits of the left operand to the left by the number of positions specified in the right operand.

`>>` (Right shift): Shifts the bits of the left operand to the right by the number of positions specified in the right operand.

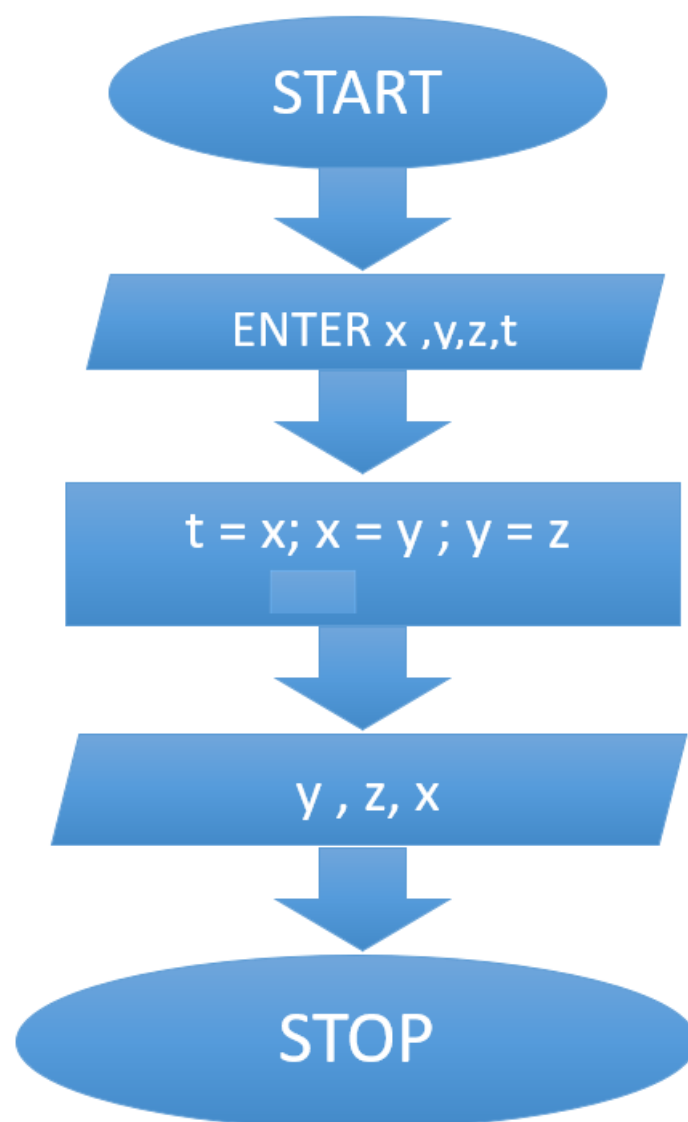
### **Comma Operator:**

`,` (Comma): Separates expressions and evaluates them from left to right. The value of the entire expression is the value of the rightmost expression.

These are some of the fundamental operators in C programming that allow you to perform various operations on variables and values.

**Q2 (a) Given the values of the variables x, y and z, write a program to rotate their values such that x has the value of y, y has the value of z, and z has the value of x.**

FLOWCHART:



## SOURCE CODE:

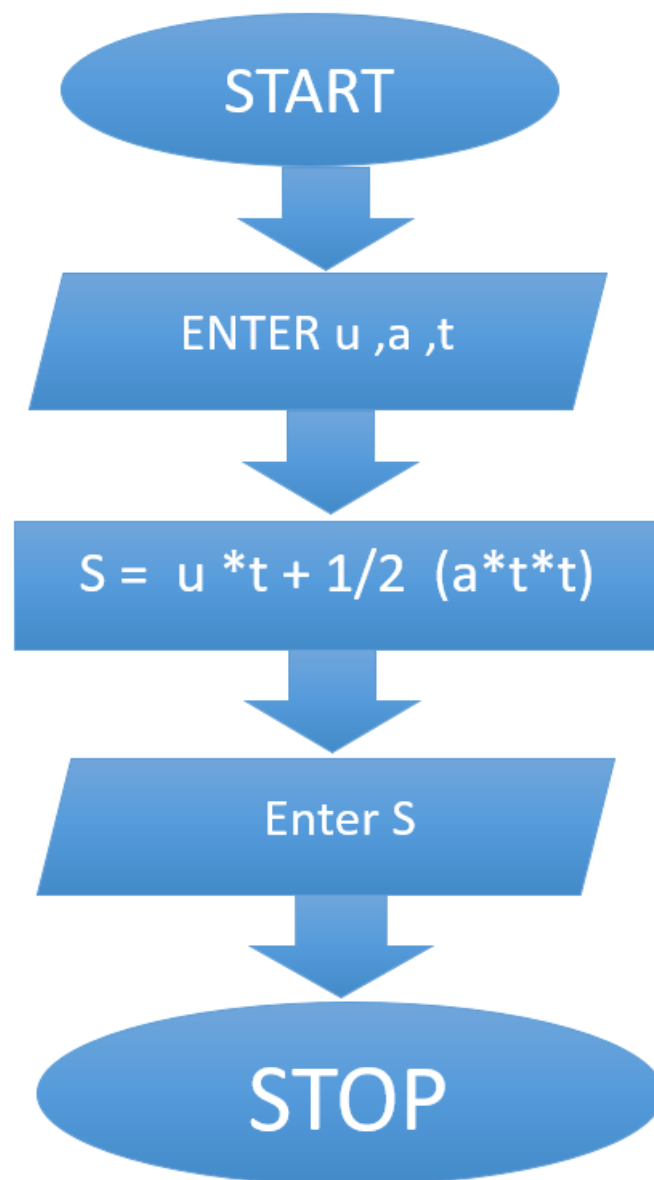
```
#include<stdio.h>
int main()
{
    int x,y,z,t;
    printf("NAME: PUSHKAR KANJANI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUES OF x,y,x");
    scanf("%d", &x);
    scanf("%d", &y);
    scanf("%d", &z);
    t=x;
    x=y;
    y=z;
    z=t;
    printf("x=%d,y=%d,z=%d",x,y,z);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUES OF x,y,x
3 5 9
x=5,y=9,z=3
```

(b) The total distance travelled by a vehicle in  $t$  seconds is given by  $\text{distance} = ut + \frac{at^2}{2}$  where  $u$  is the initial velocity (metres per second),  $a$  is the acceleration (metres per second<sup>2</sup>). Write a program to evaluate the distance travelled at regular intervals of time, given the values of  $u$  and  $a$ . The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of  $u$  and  $a$ .

FLOWCHART:



## SOURCE CODE:

```
#include<stdio.h>
int main()
{
    float a,u,t,d;
    printf("NAME: PUSHKAR KANJANI \nROLL NUMBER: 23TBIT177");
    printf("\nENTER VALUES OF u,a and t");
    scanf("%f %f %f",&u,&a,&t);
    d=u*t+(a*t*t)/2;
    printf("%.2f meters is the distance travelled by the
vehicle",d,u,a,t);
    return 0;
}
```

## OUTPUT:

```
NAME: PUSHKAR KANJANI
ROLL NUMBER: 23TBIT177
ENTER VALUES OF u,a and t
15 5 2
40.00 meters is the distance travelled by the vehicle
```