

1. Write a program to convert a given character from lower-case to upper case & vice versa

Algorithm

Step : 1 Start

Step : 2 Declare variable ch.

Step : 3 Input the character

Step : 4 Check if the input character is lowercase or uppercase.

Step : 5 If ch is in uppercase go to step 6 else go to step 7.

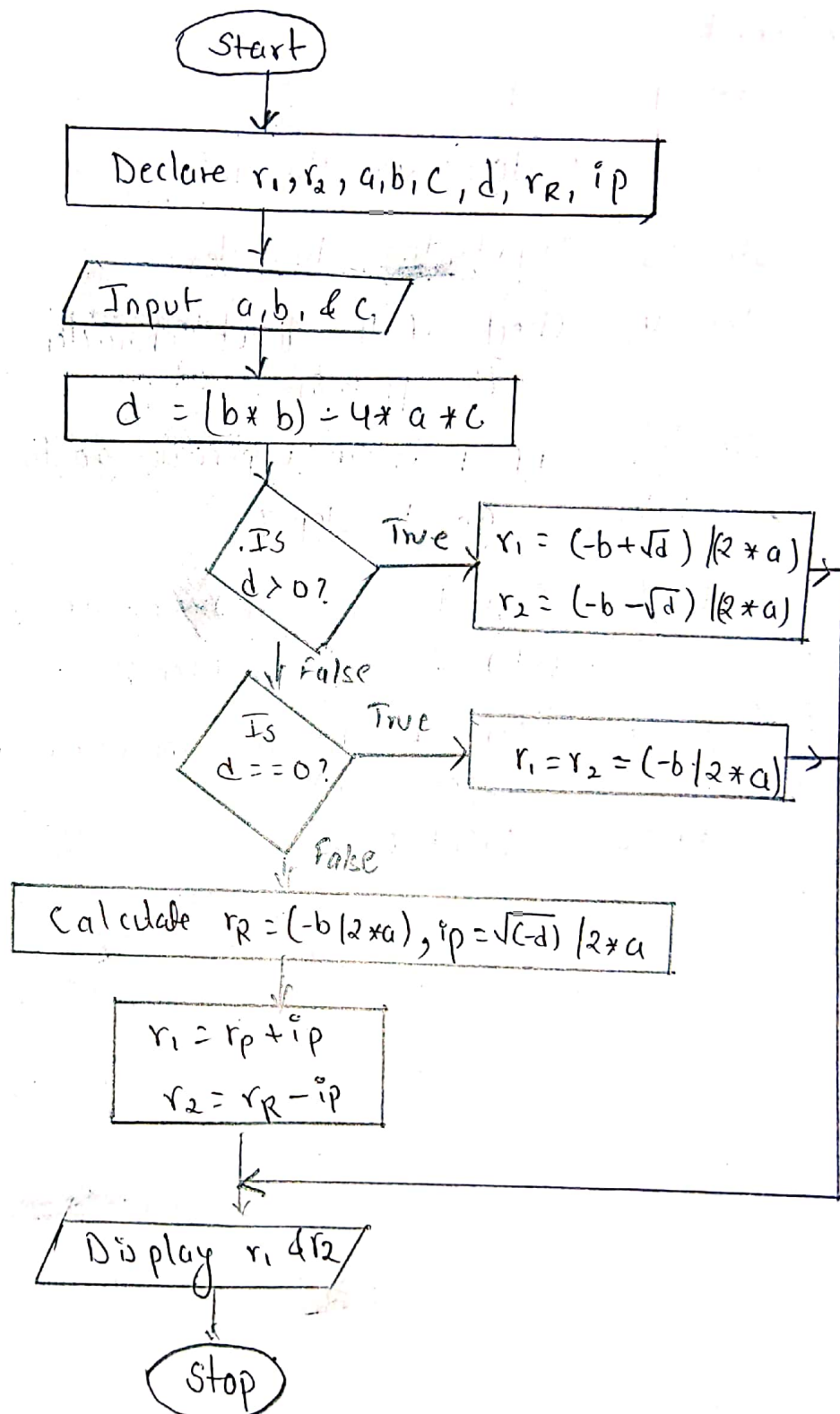
Step : 6 Convert ch in lowercase ie ch = to lower (ch) & go to step 8

Step : 7 Convert ch in uppercase ie ch = to upper (ch).

Step : 8 Print ch

Step : 9 Stop.

Flow chart to find roots of quadratic eqⁿ.



2. K1AP to print roots of a quadratic eqn (both real & imaginary).

Algorithm:

Step 1: Start

Step 2: Declare $r_1, r_2, a, b, c, d, r_r, ip$

Step 3: Input $a, b, \& c$.

Step 4: Calculate d as $d = (b \times b) - 4 \times a \times c$

Step 5: If $d > 0$, go to step 5.1 else goto step 6.

Step 5.1: Calculate r_1 as $r_1 = (-b + \sqrt{d}) / (2 \times a)$.

Step 5.2: Calculate r_2 as $r_2 = (-b - \sqrt{d}) / (2 \times a)$ and go to step 8.

Step 6: If $d = 0$, go to step 6.1 else go to step 7.

Step 6.1: Calculate $r_1 \& r_2$ as $r_1 = r_2 = (-b / 2 \times a)$ & go to step 8.

Step 7: If $d < 0$ go to step 7.1 else go to step 8

Step 7.1: Calculate real part as $r_r = -b / (2 \times a)$

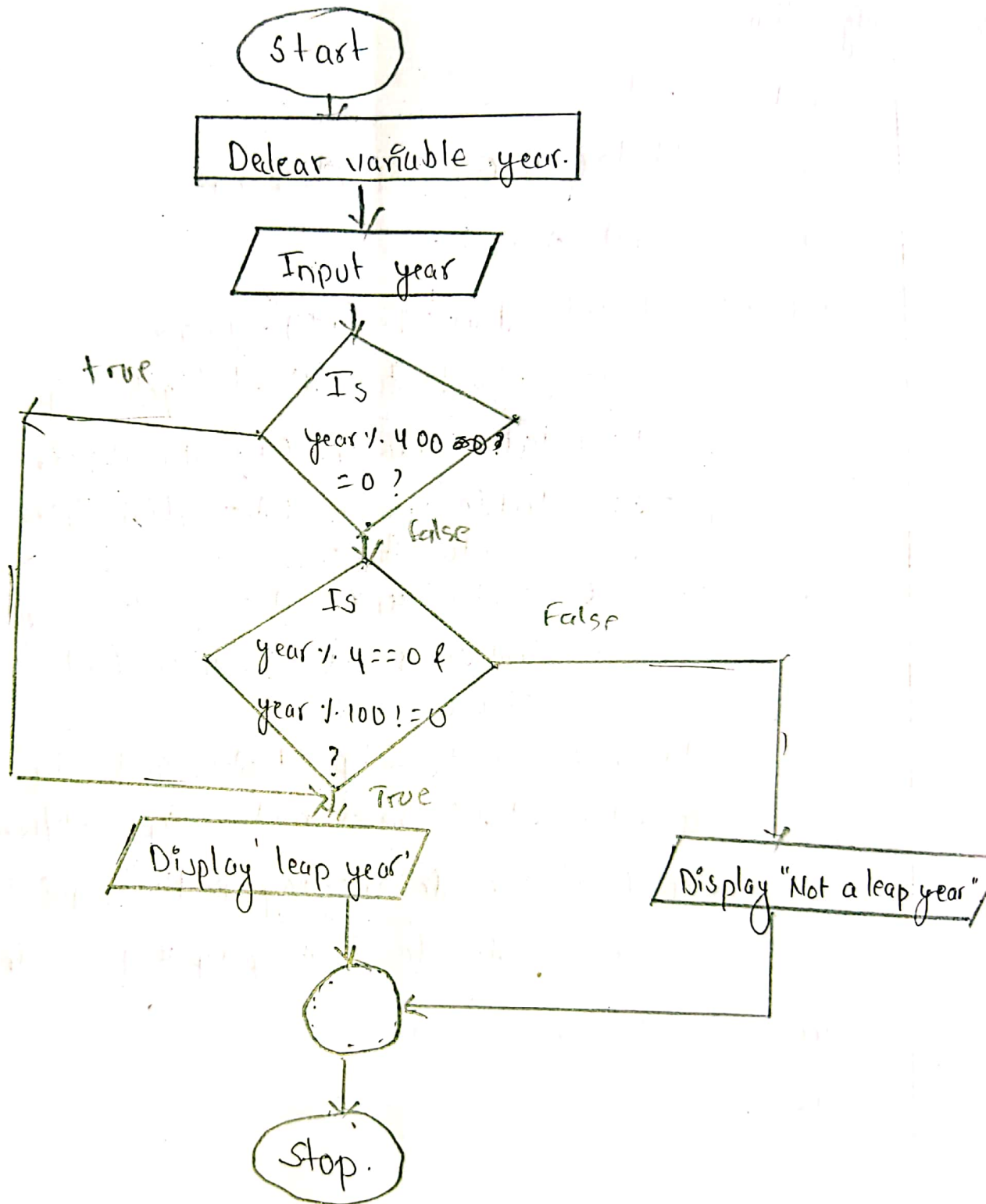
Step 7.2: Calculate imaginary part as $ip = \sqrt{d} / (2 \times a)$

Step 7.3: Calculate r_1 as $r_r + ip$ & r_2 as $r_r - ip$

Step 8: Display $r_1 \& r_2$

Step 9: Stop

Flowchart to find whether a year is leap year or not.



3. WAP to check whether a year is leap year or not.

Step 1: Start

Step 2: Declare variable, year

Step 3: Input year

Step 4: Check, if $\text{year} \bmod 400 = 0$ as $\text{year} \% 400 = 0$, go to step 6 else go to step 5.

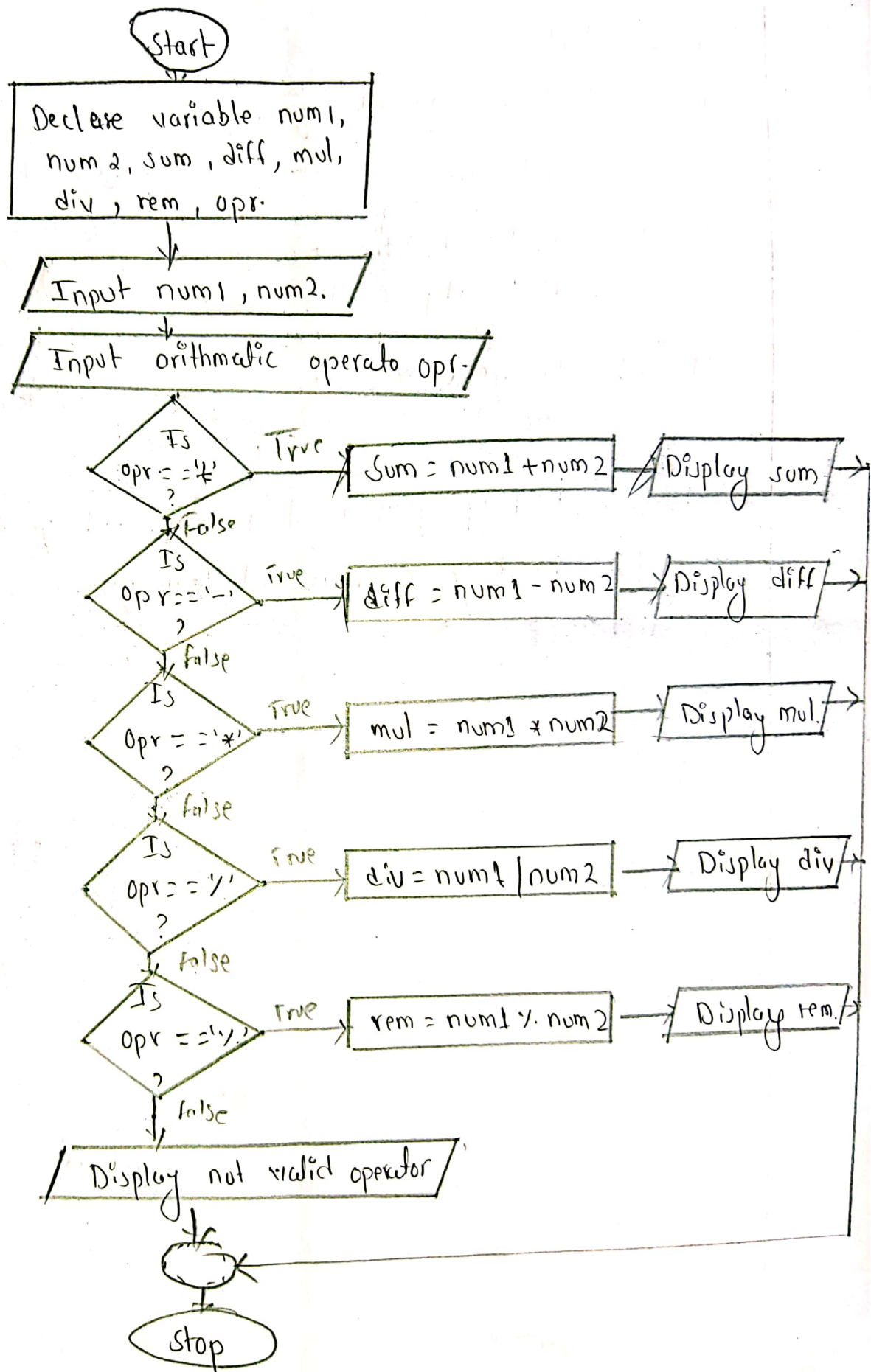
Step 5: If $\text{year} \bmod 4 = 0$ & $\text{year} \bmod 100 \neq 0$ then go to step 6 else go to step 7.

Step 6: Display year is leap year and go to step 8.

Step 7: Display year is not leap year.

Step 8: Stop.

Flowchart to show arithmetic operators by switch case



4. KAP that read two no. & an. arithmetic operator (+, -, *, /) & perform the operation as per the operator supplied using switch case.

Step 1 : Start

Step 2 : Declare variable num1, num2, sum, mul, diff, div, rem, opr.

Step 3 : Input num1 & num2

Step 4 : Input the arithmetic operator in opr

Step 5 : If opr = '+' go to step 5.1 else go to step 6.

Step 5.1 : calculate sum as $sum = num1 + num2$

Step 5.2 : Display sum & go to step 11

Step 6 : If opr = '-' go to step 6.1 else go to step 7

Step 6.1 : Calculate diff as $diff = num1 - num2$

Step 6.2 : Display diff & go to step 11

Step 7 : If opr = '*' go to step 7.1 else go to step 8

Step 7.1 : Calculate mul as $mul = num1 * num2$.

Step 7.2 : Display mul & go to step 11

Step 8 : If opr = '/' go to step 8.1 else go to step 9.

Step 8.1 : calculate div as $div = num1 / num2$.

Step 8.2 : Display div & go to step 11.

Step 9 : If opr = '%' go to step 9.1 else go to step 10

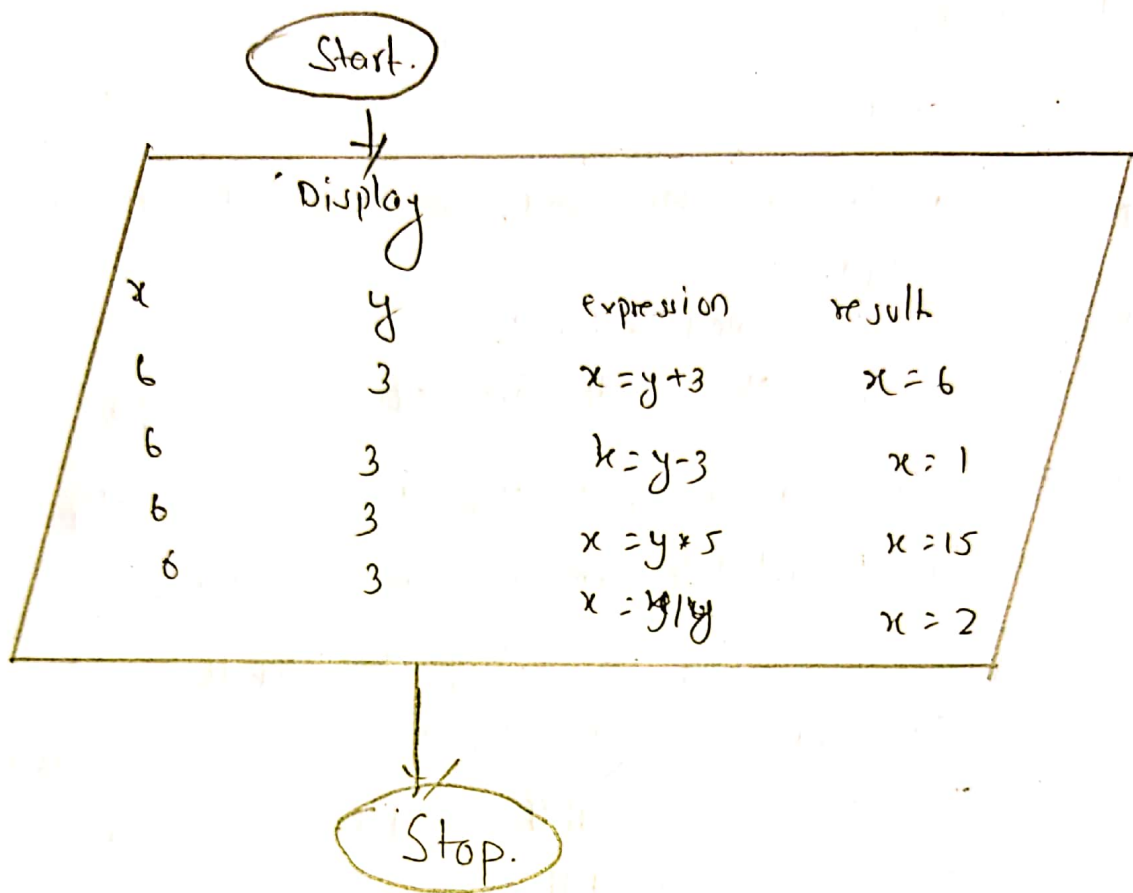
Step 9.1 : calculate rem as $rem = num1 \% num2$.

Step 9.2 : Display rem & go to step 11.

Step 10 : Display invalid operator & go to step 11

Step 11 : Stop

Flowchart to display output as



5) To display the outputs as,

x	y	expression	result.
6	3	$x = y + 3$	$x = 6$
6	3	$x = y - 2$	$x = 1$
6	3	$x = y * 5$	$x = 15$
6	3	$x = x / 2$	$x = 2$

Step 1 : start

Step 2 : Declare variable x, y.

Step 3 : Initialize $x = 6$ & $y = 3$

Step 4 : Execute & display the following.

Step 4.1 : x | t | t | y | t | t expression | t | t | x = value of (x) | n.

Step 4.2 : 6 | t | t | 3 | t | t $x = y + 3$ | t | t | t | x = value of (x) | n

Step 4.3 : 6 | t | t | 3 | t | t $x = y - 2$ | t | t | t | x = value of (x) | n

Step 4.4 : 6 | t | t | 3 | t | t $x = y * 5$ | t | t | t | x = value of (x) | n

Step 4.5 : 6 | t | t | 3 | t | t $x = x / 2$ | t | t | t | x = value of (x) | n.

Step 5 : stop

~~Source code~~

1. WAP to convert a given character from lower case to upper case. & vice versa

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

```
#include <conio.h>
```

```
int main() {
```

```
    char ch;
```

```
    printf("Enter a character ");
```

```
    scanf("%c", &ch);
```

```
    if ((ch >='A' & ch <='Z')
```

```
{
```

```
        printf("%c is in upper case\n", ch);
```

```
        ch = tolower(ch);
```

```
        printf("Its eqv lower case is %c\n", ch);
```

```
}
```

```
    else if (ch >='a' & ch <='z')
```

```
{
```

```
        printf("%c is in lower case\n", ch);
```

```
        ch = toupper(ch);
```

```
        printf("%c is eqv upper case is %c\n", ch);
```

```
}
```

```
    else
```

```
{
```

```
        printf("Enter valid value\n");
```

```
}
```

```
    return 0;
```

```
}
```

Output

Enter a character : c

Its eqv upper case is C.

2. WAP to print root of a quadratic eqⁿ (both real & imaginary)

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

```
#include <conic.h>
```

```
void main ( ) {
```

```
float a, b, c, d, r1, r2, rp, ip;
```

```
printf ("enter coefficient of a, b & c;");
```

```
scanf ("%f %f %f", &a, &b, &c);
```

```
d = pow(b, 2) - 4 * a * c;
```

```
if (d > 0)
```

```
{
```

```
r1 = (-b + sqrt(d)) / (2 * a);
```

```
r2 = (-b - sqrt(d)) / (2 * a);
```

```
printf ("The root of %f & %f, ", r1, r2);
```

```
}
```

```
else if (d == 0)
```

```
{
```

```
r1 = r2 = -b / (2 * a);
```

```
printf ("root 1 = root 2 = %f, ", r1);
```

```
}
```

```
else
```

```
{
```

```
rp = -b / (2 * a);
```

```
printf ("root 1 = root 2 = %f & ", rp);
```

```
}
```

```
else
```

```
{
```

```
rp = -b / (2 * a);
```

```
ip = sqrt(-d) / (2 * a);
```

```
printf ("The roots are : root 1 = %f + %f i /n & root 2 = %f - %f i",
```

```
rp, ip, rp, ip);
```

```
}
```

```
return 0;
```

```
}
```

output

Enter coefficient of a, b, c = 2, 9, 10

The roots are : root 1 = -0.56 + 0.97i

& root 2 = -0.56 - 0.97i.

3) WAP to check whether a year is leap year or not.

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int year;
    printf("Enter a year:");
    scanf("%d", &year);
    if (year % 400 == 0)
    {
        printf("%d is a leap year, year);
    }
    else if (year % 100 != 0 & year % 4 == 0);
    {
        printf("%d is a leap year", year);
    }
    else
    {
        printf("%d is not a leap year", year);
    }
    getch();
}
```

Output

Enter a year : 2020

2020 is leap year.

4. Write a program to read two no. on arithmetic operator (+, -, /, *, %) & perform the operation as per the operation given by the user.

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int num1, num2, sum, diff, mul, div, rem;
    char opr;
    printf("Enter two no. : ");
    scanf("%d %d", &num1, &num2);
    printf("Enter operator ");
    flush(stdin);
    opr = getch();
    switch (opr)
    {
        case '+':
            sum = num1 + num2;
            printf("Sum is %d", sum);
            break;
        case '-':
            diff = num1 - num2;
            printf("Difference is %d", diff);
            break;
        case '*':
            mul = num1 * num2;
            printf("Product is %d", mul);
            break;
        case '/':
            div = num1 / num2;
            printf("Division is %d", div);
            break;
        case '%':
            rem = num1 % num2;
            printf("Remainder is %d", rem);
            break;
        default:
            printf("Enter a valid operator");
    }
    getch();
}
```

Output
Enter two no. :
8
100
Enter operator : *
Product is 800.


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

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

void main() {

```
printf("text | type | it expression | it result");
```

```
printf("ln 1t61t1t31t11t x=y+3 1t7 1t6 ");
```

```
printf("\n\t6\t|\t1\t|\t3\t|\t1\t| x=y-2 \t|\t1\t|\t1");
```

```
printf("In | t6 | t1 \ t3 \ t1 \ t7 * y * 3 \ t1 \ t15");
```

```
printf("\n\t6\t|\t1\t|\t3\t|\t1\t|\tx=x/y\t|\t1\t|\tt2");
```

geteilt);

3

x		y		expression		result.
6	1	3	1	$x = y + 3$	1	6
6	1	3	1	$x = y - 2$	1	1
6	1	3	1	$x = y * 3$	1	15
6	1	3	1	$x = x / y$	1	2

Discussion & conclusion.

In lab, we were able to use our theoretical knowledge of branching / control statement. like : if statements, if -- else statements, nested if -- else statement, else -- if ladder statement and switch statement. We learned to write the codes . step by step in order to solve a problem.

We also learned to give proper structure to a program & execute & display a proper output.