EXPERIMENT NO.: 06

6.1 Objective: To evaluate the solution of fourth order differential equation by R K method using C/C++.

6.2 Theory: The solution of the differential equation of first order $\frac{d}{d} = f(,)$ is given by the formula:

k_1= h x f(x_0 , y_0)

k_2= h x $f(_0 + \frac{h}{2}, _0 + \frac{k_1}{2})$

k_3= h x $f(_0 + \frac{h}{2}, _0 + \frac{k_2}{})$

k_4= h x $f(_0 + h, _0 + k_3)$

$\Delta y = \frac{1}{6}$ x [k_1 + 2(k_2 + k_3) + k_4]

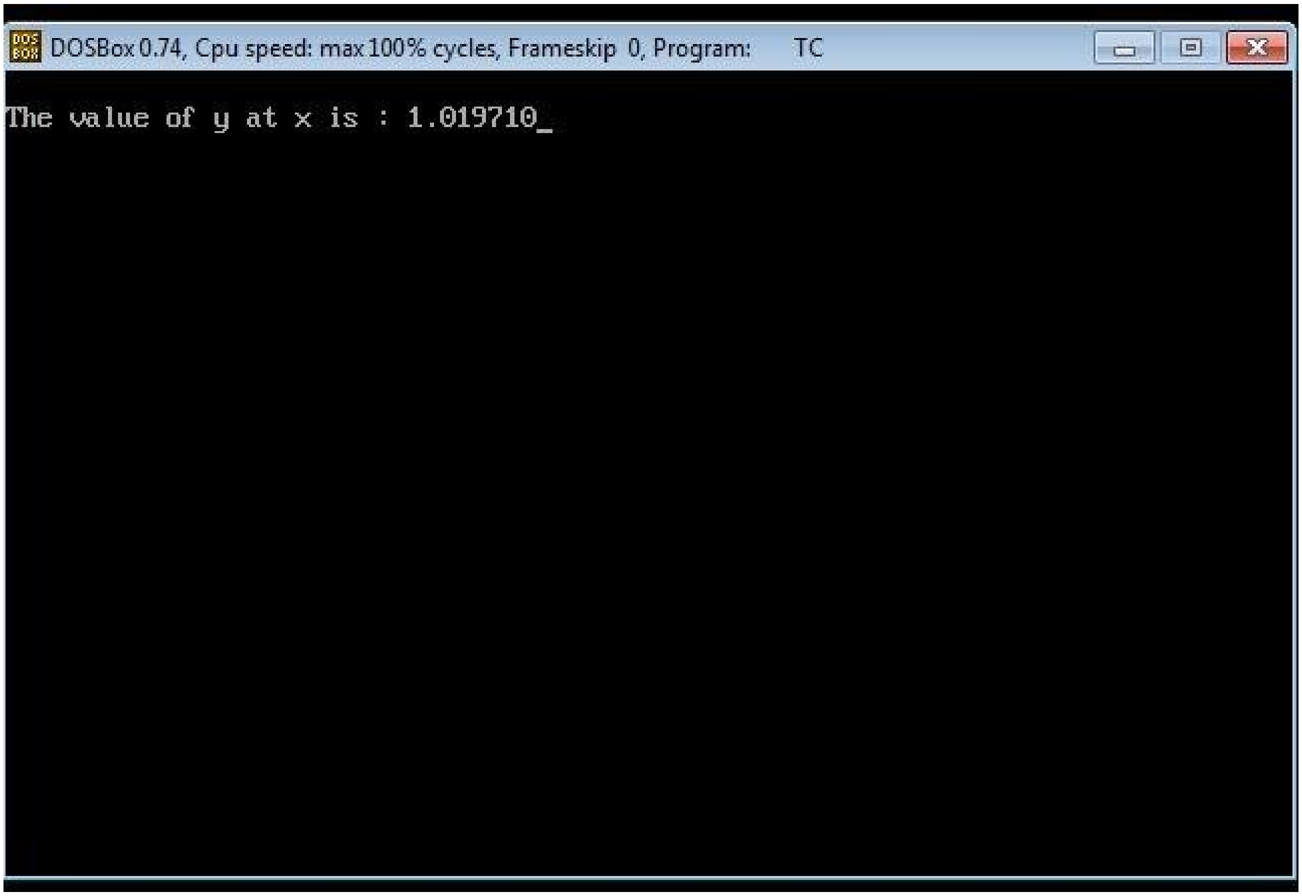
$Y_1 = y_0 + \Delta y$

6.3 Procedure/Code:

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

float dydx(float x, float y)

{

   return((x - y)/2);

}

float RungeKutta(float  x0, float y0, float x, float h)

{

   int n = (int)((x - x0) / h);

   float k1, k2, k3, k4, k5;
```

```
float y = y0;
for (int i=1; i<=n; i++)
{
    k1 = h*dydx(x0, y);
    k2 = h*dydx(x0 + 0.5*h, y + 0.5*k1);
    k3 = h*dydx(x0 + 0.5*h, y + 0.5*k2);
    k4 = h*dydx(x0 + h, y + k3);
    y = y + (1.0/6.0)*(k1 + 2*k2 + 2*k3 + k4);;
    x0 = x0 + h;
}
return y;
}
int main()
{
clrscr();
float x0 = 0, y = 1, x = 2, h = 0.2;
printf("\nThe value of y at x is : %f", RungeKutta(x0, y, x, h));
return 0;
getch();
}
```

6.4 Output:


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
The value of y at x is : 1.019710_

EXPERIMENT NO.: 05

5.1 Objective: To evaluate the solution of first order differential equation by Euler's Method using C/C++.

5.2 Theory: The solution of the differential equation of first order $\frac{d}{d} = f(,)$ is given by the formula: $\boxed{{}_{\boldsymbol{n+1}} = {}_{n} + f({}_{n}, {}_{n})}$

5.3 Procedure/Code:

```
#include<iostream.h>
#include<conio.h>
#include<math.h> using
namespace std; float
func(float x, float y)
{

   return (x + y + x * y);
}
void Euler(float x0, float y, float h, float x)
{
   float temp = -0;
   while (x0 < x)
   {
           temp = y;
           y = y + h * func(x0, y);
           x0 = x0 + h;
   }
   cout << "Approximate solution at x = "
           << x << " is " << y << endl;
}
int main()
{
   clrscr();
   float x0 = 0;
   float y0 = 1;
   float h = 0.025;
   float x = 0.1;
```

```
    Euler(x0, y0, h, x);
    getch();
    return 0;
}
```

5.4 Output: