EXPERIMENT-8

AIM-Solution of initial value problem using Euler's Method

THEORY-

In mathematics and computational science, the Euler method (also called forward Euler method) is a first-order numerical procedure for solving ordinary differential equations (ODEs) with a given initial value.

Consider a differential equation **dy/dx = f(x, y)** with initial condition **y(x0)=y0**

then succesive approximation of this equation can be given by:
y(n+1) = y(n) + h * f(x(n), y(n))

where h = (x(n) - x(0)) / n
h indicates step size.

ALGORITHM-

1. Enter the initial values of x and y(x0 and y0).
2. Enter the value of x, for which y is to be determined.
3. Enter the width of the interval, 'h'.
4. Do:
   y=y0+(h*dy/dx(x0,y0))
   y0=y.
   x0=x0+h
   Until (x0>=x)
5. Print y, which is the solution.

EXPERIMENT-8

CODE-

```
#include <iostream>
using namespace std;

float func(float x, float y) // dy/dx=(x + y + xy)
{
return (x + y + x * y);
}
void euler(float x0, float y, float h, float x)
{
float temp = -0;
while (x0 < x)// Iterating till the point at which we need approximation
{
temp = y;
y = y + h * func(x0, y);
x0 = x0 + h;
}
cout << "Approximate solution at x = "
<< x << " i.e\n"<<" y("<<x<<") = " << y << endl; // Printing approximation
}
int main()
{
float x0 = 0;
float y0 = 1;
float h = 0.025;
float x = 0.1; // Value of x at which we need approximation
euler(x0, y0, h, x);
return 0;
}
```

OUTPUT=