Ministerul Educaţiei al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare Informatica si Microelectronica

RAPORT

Lucrarea de laborator nr.4
Disciplina: *Metode numerice*

A efectuat:
st. gr. CR-191 Frecventa Redusa Balan Ion

A verificat:
lect., sup. A.Godonoga

Chisinau 2020

Tema: Integrarea numerica a ecuatiilor diferentiale

Scopul lucrarii:

1) Sa se determine solutia problemei Cauchy

$$\frac{dy}{dx} = f(x, y)$$

$$y(a) = b$$

pe segmentul indicat [a,b] prin metodele Euler, Euler modificat si Runge-Kutta cu pasul 0.05;

2) Sa se efectueze o analiza a rezulatelor obtinute

Notiuni teoretice:

Metoda Euler

Conform principiului de baza, la fiecare iteratie functia se aproximeaza prin adaugarea la valoarea curenta a unei corectii egala cu produsul dintre pasul de integrare si derivata functiei. Din punct de vedere formal, acest principiu descrie de fapt *metoda Euler*:

$$y_{k+1} = y_k + h \cdot f(x_k, y_k)$$

Originea acestei formule se gaseste insa in dezvoltarea in serii Taylor a functiei $y(x)$ in jurul punctului x_k:

$$y(x) = y(x_k) + \frac{x - x_k}{1!} y'(x_k) + \frac{(x - x_k)^2}{2!} y''(x_k) + \ldots$$

Daca se cunoaste valoarea functiei in punctul x_k, notata $y_k = y(x_k)$, functia $f(x,y)$ ce defineste ecuatia diferentiala ce defineste problema Cauchy permite calculul derivatelor de orice ordin $y^{(q)}(x)$ care apar in ultima relatie.
Prin urmare, daca se calculeazaderivatele $y^{(q)}(x)$, este posibila evaluarea functiei $y(x)$ in orice punct din vecinatatea lui x_k. Aceasta este o metoda numerica, cunoscuta sub numele de *metoda Taylor*. In cazul in care se considera o distributie uniforma a punctelor x_k ($x_{k+1} = x_k + h$) este posibil ca pornind din punctul x_0 corespunzator conditiei initiale a problemei Cauchy si folosind formula:

$$y_{k+1} = y_k + \frac{h}{1!} y'_k + \frac{h^2}{2!} y''_k + \ldots + \frac{h^p}{p!} y_k^{(p)} + \ldots$$

sa se determine solutia numerica a ecuatiei diferentiale sub forma unui sir de puncte (x_0, y_0), (x_1, y_1), ... , (x_k, y_k), ...Daca din ultima relatie se retin numai primii doi termeni, se regaseste formula corespunzatoare *metodei Euler*:

$$y_{k+1} \approx y_k + \frac{h}{1!} y'_k = y_k + h \cdot f(x_k, y_k)$$

Metoda Euler modificata

Considerăm pe [xi, xi+1] ca direcţie a segmentului MiMi+1 direcţia definită de punctul de la mijlocul segmentului (nu de extremitatea stângă ca în formula iniţială) se obţine metoda Euler modificată. Dacă xi, yi sunt valori calculate, procesul iterativ este următorul:

$$\begin{cases} x_i = x_0 + ih;\ f_i = f(x_i, y_i);\ x_{i+\frac{1}{2}} = x_i + \frac{h}{2}; \\ y_{i+\frac{1}{2}} = y_i + \frac{h}{2} f_i;\ f_{i+\frac{1}{2}} = f\left(x_{i+\frac{1}{2}}, y_{i+\frac{1}{2}}\right);\ y_{i+1} = y_i + hf_{i+\frac{1}{2}}; \end{cases}$$

Pentru această metodă $\tau_i(h)$ este de ordinul $O(h^2)$.

Metodele Runge-Kutta

In cazul metodei Euler obisnuite deplasarea intre doua puncte de calcul succesive x_k si x_{k+1}, se face prin aplicarea unei corectii valorii y_k determinata de produsul pasului de integrare *h* si derivata solutie *y(x)* calculata la extremitatea stanga a intervalului y_k, folosind functia *f* ce defineste ecuatia diferentiala. Se poate scrie, asadar:

$$K_1 = h \cdot f(x_k, y_k)$$
$$y_{k+1} = y_k + K_1$$

Pentru versiunea Cauchy, imbunatatirea preciziei se obtine calculand corectia ce se aplica lui y_k in functia de derivata lui *y(x)* la mijlocul intervalului de lucru $x_{k+1/2}=x_k+h/2$. Calculul derivatei in acest punct necesita o aproximare pentru $y_{k+1/2}$, care se obtine folosind un pas Euler clasic, adica:

$$K_1 = h \cdot f(x_k, y_k)$$
$$K_2 = h \cdot f(x_k + \frac{h}{2}, y_k + \frac{K_1}{2})$$
$$y_{k+1} = y_k + K_2$$

Metodele Runge–Kutta generalizeaza acest principiu. Pornind de la un punct de coordonate cunoscute (x_k,y_k), metodele Runge–Kutta avanseaza la punctul urmator aplicand valorii y_k o combinatie liniara de corectii K_j , ponderate de o serie de coeficienti g_{rj} ce urmeaza a fi determinati:

$$y_{k+1} = y_k + \sum_{j=1}^{r} \gamma_{rj} \cdot K_j$$

In aceasta relatie *r* indica ordinul metodei. Metodele de tip Runge–Kutta au o serie de avantaje dintre care se amintesc: *(i)* sunt metode directe, deci nu necesita folosirea unor metode auxiliare la pornire; *(ii)* sunt identice cu seriile Taylor pana la termenul de rang *r*, deci exista posibilitatea estimarii erorii de trunchiere; *(iii)* nu necesita evaluari ale derivatelor partiale ale functiei *f(x,y)*, ci numai ale functiei *f* in sine.

Corectiile K_j se determina ca produs al pasului de integrare *h* cu anumite valori ale functiei *f*, calculate in puncte din vecinatatea punctului (x_k, y_k):

$$K_j = h \cdot f(\xi_j, \eta_j)$$
$$\xi_j = x_k + \alpha_j \cdot h \qquad j = 1, \dots, r$$
$$\eta_j = y_k + \sum_{i=1}^{j-1} \beta_{ij} \cdot K_i$$

In toate cazurile se considera:

$\alpha_1 = 0$ $\quad$ $\beta_{11} = 0$ $\quad$ deci: $\quad$ $\xi_1 = x_k$ $\quad$ $\eta_1 = y_k$

astfel incat se poate scrie:

$$K_1 = h \cdot f(x_k, y_k)$$
$$K_2 = h \cdot f(x_k + \alpha_2 \cdot h, y_k + \beta_{21} \cdot K_1)$$
$$K_3 = h \cdot f(x_k + \alpha_3 \cdot h, y_k + \beta_{31} \cdot K_1 + \beta_{32} \cdot K_2)$$
$$s.a.m.d.$$

$y' = e^x - y^2$, $y(0)=0$; $a=0$; $b=1$;

Codul programului

```
#include <iostream>
#include <cmath>
using namespace std;
float f(float(x), float(y)){
	return ((exp(x)) - pow(y,2));
}

int main(){
	int n, i;
	double a, b;
	float h;
	float k0[25], k1[25], k2[25], k3[25];
	cout << " Ecuatia dy/dx = (exp(x) - pow(y,2))"<< endl;
	cout << " Introduceti intervalul:" << endl << " a = ";
	cin >> a;
	cout << " b = ";
	cin >> b;
	cout << " Introduceti pasul: ";
	cin >> h;
	n = (b - a) / h;
	double y[10], g[10], x[10], Y[10], L[10];
	cout << " Introduceti x0: ";
	cin >> x[0];
	cout << " Introduceti y0: ";
	cin >> y[0];
	cout << " --------------------------------------------" << endl;
	cout << " Metoda Euler " << endl;
	cout << " --------------------------------------------" << endl;
	for (i = 1; i <= n; i++) {
		x[i] = x[i - 1] + h;
	}
	for (i = 1; i <= n; i++){
		y[i] = y[i - 1] + (h*f(x[i - 1], y[i - 1]));
	}

	cout << " Iteratii    x        y          f(x,y)" << endl;
	for (i = 1; i <= n; i++) {
		cout << "   " << i << "\t " << x[i] << "\t " << y[i] << "\t  " <<
f(x[i], y[i]) << endl;
	}
	cout << " --------------------------------------------" << endl;
	cout << " Metoda Euler Modificata " << endl;
	cout << " --------------------------------------------" << endl;
	for (i = 1; i <= n; i++) {
		x[i] = x[i - 1] + h;
	}
	for (i = 1; i <= n; i++){
		y[i] = y[i - 1] + (h*f(x[i - 1], y[i - 1]));
		y[i] = y[i - 1] + ((h / 2)*(f(x[i - 1], y[i - 1]) + f(x[i],
y[i])));
	}
	cout << " Iteratii    x        y          f(x,y)" << endl;
	for (i = 1; i <= n; i++) {
		cout << "   " << i << "\t " << x[i] << "\t " << y[i] << "\t  " <<
f(x[i], y[i]) << endl;
```

```
        }
        cout << " --------------------------------------------" << endl;
        cout << " Metoda Runge-Kutta " << endl;
        cout << " --------------------------------------------" << endl;
        for (int i = 1; i <=n; i++)
        {
               k0[i - 1] = f(x[i - 1], y[i - 1]);
               k1[i - 1] = f(x[i - 1] + 0.5*h, y[i - 1] + 0.5*k0[i - 1]);
               k2[i - 1] = f(x[i - 1] + 0.5*h, y[i - 1] + 0.5*k1[i - 1]);
               k3[i - 1] = f(x[i - 1] + h, y[i - 1] + k2[i - 1]);
               y[i] = y[i - 1] + h / 6 * (k0[i - 1] + 2 * k1[i - 1] + 2 * k2[i -
1] + k3[i - 1]);
        }
        cout << " Iteratii    x        y           f(x,y)" << endl;
        for (i = 1; i <= n; i++) {
               cout << "   " << i << "\t " << x[i] << "\t " << y[i] << "\t  " <<
f(x[i], y[i]) << endl;
        }
        cin.get();
        cin.get();
        return 0;
}
```

Rezultatele obtinute

```
options   compilation   execution

Ecuatia dy/dx = (exp(x) - pow(y,2))
Introduceti intervalul:
a = 0
b = 1
Introduceti pasul: 0.5
Introduceti x0: -1
Introduceti y0: 0
--------------------------------------------
Metoda Euler
--------------------------------------------
Iteratii    x        y           f(x,y)
  1      -0.5     0.18394        0.572697
  2      0        0.470288       0.778829
--------------------------------------------
Metoda Euler Modificata
--------------------------------------------
Iteratii    x        y           f(x,y)
  1      -0.5     0.235144       0.551238
  2      0        0.557734       0.688933
--------------------------------------------
Metoda Runge-Kutta
--------------------------------------------
Iteratii    x        y           f(x,y)
  1      -0.5     0.210003       0.56243
  2      0        0.473306       0.775981
```

Fig.2 Metoda Euler,Runge-Kutta

Concluzii

Efectuind aceasta lucrare de laborator am isusit cum se determina solutia problemei Cauchy prin diferite metode :

1) metoda Euler,
2) metoda Euler modificata,
3) metoda Runge-Kutta.

Aceste metode s-au obtinut cu ajutorul dezvoltarii seriei Taylor :
$y(x)=y(x_0)+y'(x_0)h+1/2y''(x_0)h^2+1/3y'''(x_0)h^3+1/4y''''(x_0)h^4+1/5y'''''(x_0)h^5+\ldots$ Pentru prima metoda s-au pastrat primii 2 termeni a dezvoltarii si respectiv, eroarea e proportionala cu h^2, in a doua metoda s-au pastrat primii 3 termeni si s-a obtinut eroarea proportionala cu h^3, iar pentru a treia metoda, s-au pastrat primii 5 termeni a dezvoltarii si s-a obtinut o eroare proportionala cu h^5.