

Projekt iz predmeta *“Numeričke metode”*

Student: Edin Smajić

Smjer: Teorijska kompjuterska nauka (TKN)

Godina: Druga (II)

Status: Redovni-samofinansirajući

Broj indeksa: 5810/M

Tema projekta:

Napraviti kompjuterski program za rješavanje
diferencijalnih jednačina i analizirati njegovu stabilnost
na raznim primjerima

Uvod:

U prilogu sa ovom dokumentacijom poslat je i folder pod imenom "Projekt" u kojem se nalaze: "projekt.cbp", "projekt.h", "projekt.cpp" i "main.cpp";

"projekt.cbp" - fajl koji pri pokretanju automatski otvara ostale navedene fajlove.

"projekt.h" - header fajl, u kojem su navedeni prototipovi funkcija, bez implementacija istih, te su u njemu uključeni neki drugi headeri koji su neophodni za ispravan rad programa.

"projekt.cpp" - source fajl, u kojem su implementirane sve funkcije čiji su prototipovi navedeni u fajlu "projekt.h".

"main.cpp" - source fajl, u kojem se vrše testiranja tih funkcija, o kojima će ispod biti više govora.

U navedenom "projekt.cpp" fajlu implementirao sam Runge-Kutta metodu četvrtog reda za rješavanje običnih diferencijalnih jednačina, kojem je totalna akumulirana greška $O(h^4)$. Preciznije implementirao sam 2 različite funkcije, koje će biti objašnjene nešto kasnije.

Formulacija problema rješavanja diferencijalnih jednačina

- Treba da nađemo funkciju $y(x)$, $x \in [a, b]$ takvu da ona zadovoljava diferencijalnu jednačinu prvog reda $dy/dx = y' = f(x, y)$, uz početni uslov $y(x_0) = y_0$.

Formule koje sam koristio u programu su date sljedećim slikama: (!)

$$\Phi = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$k_1 = f(x, y),$$

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

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

$$k_4 = f(x + h, y + hk_3).$$

$$y_{i+1} = y_i + h\Phi(x_i, y_i, h; f), \quad i = 0, \dots, n-1.$$

Funkcije:

Funkcije 1) i 2) su ustvari sama suština projekta, dok je funkcija 3) pomoćna.

1) void **MetodaRungeKutta1**(double(**f**)(double, double), double x0, double y0, const double &h);

Ova funkcija ima 4 parametra i to:

- Prvi parametar je funkcija **f** (konkretno pokazivač na tu novu funkciju **f**) za koju vrijedi $y' = f(x, y)$; tj. **f** jednoznačno određuje diferencijalnu jednačinu koju treba riješiti.
- Drugi i treći parametri su realnog tipa, za koje vrijedi početni uvjet/uslov diferencijalne jednačine koju treba riješiti tj. vrijedi: $y(x_0) = y_0$.
- Četvrti parametar je također realnog tipa, te on predstavlja veličinu koraka u procesu samog rješavanja diferencijalne jednačine.
- Ova funkcija na početku ispiše šablon kojeg će ispisivati funkcijom 3), te pozove funkciju 3) sa parametrima 0, x0, y0. Zatim kroz 5 iteracija mijenja x0 po koraku h, te y0 na način (!), koji je opisan u uvodu. Na kraju svake od iteracija poziva funkciju 3) sa parametrima i, x0, y0.

2) void **MetodaRungeKutta2**(double(f)(double, double), double x0, double y0, const double &x1, const int &n=10);

Ova funkcija je vrlo slična prethodnoj, samo što ima 5 parametara, od kojih je zadnji opcionalan (nije neophodan da bi funkcija radila korektno).

- Prvi, drugi i treći parametri su potpuno isti kao u funkciji 1).
- Četvrti parametar x1 je realnog tipa, za kojeg ova funkcija pronalazi konkretnu *vrijednost y1* takvu da je: $y(x1)=y1$;
- Peti parametar je opcionalan, a po defaultu je 10. Pomoću njega, ali i pomoću x0 i y0, će se izračunati korak h po formuli $h=(x1-x0)/n$.
- Ova funkcija na početku ispiše šablon kojeg će ispisivati funkcijom 3), te pozove funkciju 3) sa parametrima 0, x0, y0. Inicijalizira h, i, te epsilon. Zatim, sve dok je x0 manje od x1 mijenja x0 po koraku h, te y0 na način (!), koji je opisan u uvodu. Na kraju svake od iteracija poziva funkciju 3) sa parametrima i, x0, y0. Nakon što se vrijednost y1 nađe ispisat će se na ekran u formatu: "To y1 je: y1", međutim ako se ne pronađe ispisat će se poruka: "Nemoguće pronaći y1 sa takvim početnim uslovom".

3) void **ispis**(const double &i, const double &x0, const double &y0);

- Ovo je pomoćna funkcija koja prima 3 parametra, od kojih su sva tri realnog tipa, te ih ispisuje na ekran sa fiksiranim razmacima zbog preglednosti. Pomoćna je jer se poziva u funkcijama 1) i 2).

Literatura:

1. Predavanja od profesorice Amele Muratović-Ribić na predmetu "Numeričke metode" (Prirodno matematički fakultet Sarajevo, 2020.)
2. Numerička analiza - Zlatko Drmač, Vjeran Hari, Miljenko Marušić, Mladen Rogina, Sanja Singer, Saša Singer (Zagreb, 2003.)