Навчально-науковий комплекс "Інститут прикладного системного аналізу"

при Національному технічному університеті України "КПІ"

Факультет системних досліджень

Кафедра математичних методів системного аналізу

**Лабораторна робота №8**

**з курсу «Чисельні методи»**

Виконав:

студент групи КА-25

Поршнэва К.О.

Прийняв:

Коновалюкss М.М.

Київ – 2014 р.

**Тема :** Крайова задача другого порядку

**Умова задачі:**

Розв'язати крайову задачу

|  |  |
| --- | --- |
| ***Ay = f;*** | ***x1< x < x2;*** |
| ***a1 y = f1;*** | ***x = x1;*** |
| ***a2 y = f2;*** | ***x = x2;*** |

Де

***Ay = y'' + p(x) y' + q(x) y = f(x);***

***p(x), q(x) –*** задані функції

***a1 , а2***  - задані оператори

***x1 ,х2 -*** задані числа

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | e |
| -1.150 | -1.702 | 0.662 | -2.321 | 2.136 |

***f(x) = y” + 2xy’ – 2y ;***

***p(x)= 2x, q(x)= -2 ;***

***а1=y’(2) ,а2=0,4y(2,3)-y’(2,3) ;***

1. **Теоретичні відомості**

На відрізку  вводиться дискретна сітка ,



де вузли сітки , , .

Розв’язати задачу (1) означає знайти такий вектор , що кожне значення  наближає відповідно .  — аналітичний розв’язок задачі (1).

Обчислення першої похідної.

Вводяться різницеві похідні наступним чином:

 — перша різницева похідна “вперед”

 — перша різницева похідна “назад”

З’ясуємо точність апроксимації похідної такими різницевими похідними.





Помилка обчислення похідної:



Як бачимо, перший порядок точності. Цілком аналогічно . Помилка апроксимації .

Вводиться перша центральна різницева похідна як середнє арифметичне першої різницевої похідної “вперед” та першої різницевої похідної “назад”.

 , 

Помилка апроксимації .



Обчислення другої похідної.

Зауважимо, що перша різницева похідна “вперед” в n- му вузлі дискретної сітки є одночасно першою різницевою похідною “назад” в (n+1) – му вузлі. А перша різницева похідна “назад” в n- му вузлі дискретної сітки є одночасно першою різницевою похідною “вперед” в (n-1) – му вузлі.



 — різницева похідна “назад”;  — різницева похідна “вперед”.

Друга різницева похідна:



Помилка апроксимації другої похідної . Перевірити самостійно.



Нехай .

Враховуючи все, зазначене вище, задачу (1) запишемо наступним чином:

 (3)

це система  рівнянь з  невідомими, тобто недовизначена на 2 рівняння. Додаємо до системи 2 рівняння – рівняння крайових умов.

Наближаємо крайові умови центральними похідними:

 (\*),  (\*\*), де ,  — фіктивні змінні. Тепер розглядаємо систему (3) як систему з  () рівнянь з  невідомими:

 (3\*)

Розглянемо рівняння системи (3\*) при :

 (4)

Далі виражаємо з крайової умови (\*) змінну :  і підставляємо його в (4).

Розглянемо рівняння системи (3\*) при :

 (5)

Далі виражаємо з крайової умови (\*\*) змінну : 

і підставимо в рівняння (5). Отримали систему з  () рівнянь з  невідомими.

Результат розв’язку такої системи збігається до точного розв’язку зі швидкістю .

Отримали систему

, (6)

**причому матриця *А* є тридіагональною. Розмірність матриці ймовірно є значною  і розв’язувати її слід методом прогонки. Цей метод вимагає зберігати в пам’яті не всю матрицю системи, а лише три вектори значень.**

**Результат роботи:**

h = 0.003

X[i] y(X[i]) Y[i] e[i]

2 -7.74104 -8.18271 0.441668

2.003 -7.75885 -8.19258 0.433726

2.006 -7.77669 -8.20258 0.425887

2.009 -7.79455 -8.2127 0.418149

2.012 -7.81245 -8.22296 0.410512

2.015 -7.83036 -8.23334 0.402973

2.018 -7.84831 -8.24384 0.395532

2.021 -7.86628 -8.25447 0.388188

2.024 -7.88428 -8.26522 0.380939

2.027 -7.9023 -8.27608 0.373785

2.03 -7.92035 -8.28707 0.366724

2.033 -7.93843 -8.29818 0.359755

2.036 -7.95653 -8.30941 0.352876

2.039 -7.97466 -8.32075 0.346088

2.042 -7.99281 -8.3322 0.339388

2.045 -8.011 -8.34377 0.332776

2.048 -8.0292 -8.35545 0.326251

2.051 -8.04744 -8.36725 0.319811

2.054 -8.0657 -8.37915 0.313455

2.057 -8.08398 -8.39117 0.307183

2.06 -8.10229 -8.40329 0.300994

2.063 -8.12063 -8.41552 0.294886

2.066 -8.139 -8.42785 0.288858

2.069 -8.15739 -8.4403 0.28291

2.072 -8.1758 -8.45284 0.27704

2.075 -8.19424 -8.46549 0.271248

2.078 -8.21271 -8.47824 0.265532

2.081 -8.2312 -8.49109 0.259891

2.084 -8.24972 -8.50405 0.254326

2.087 -8.26826 -8.5171 0.248834

2.09 -8.28683 -8.53025 0.243415

2.093 -8.30543 -8.5435 0.238067

2.096 -8.32405 -8.55684 0.232791

2.099 -8.3427 -8.57028 0.227585

2.102 -8.36137 -8.58382 0.222448

2.105 -8.38006 -8.59744 0.21738

2.108 -8.39879 -8.61116 0.212379

2.111 -8.41753 -8.62498 0.207444

2.114 -8.43631 -8.63888 0.202576

2.117 -8.4551 -8.65288 0.197772

2.12 -8.47393 -8.66696 0.193032

2.123 -8.49278 -8.68113 0.188356

2.126 -8.51165 -8.69539 0.183743

2.129 -8.53055 -8.70974 0.179191

2.132 -8.54947 -8.72417 0.1747

2.135 -8.56842 -8.73869 0.17027

2.138 -8.5874 -8.75329 0.165899

2.141 -8.6064 -8.76798 0.161586

2.144 -8.62542 -8.78275 0.157332

2.147 -8.64447 -8.7976 0.153134

2.15 -8.66354 -8.81254 0.148994

2.153 -8.68264 -8.82755 0.144908

2.156 -8.70176 -8.84264 0.140878

2.159 -8.72091 -8.85782 0.136903

2.162 -8.74009 -8.87307 0.13298

2.165 -8.75928 -8.88839 0.129111

2.168 -8.77851 -8.9038 0.125294

2.171 -8.79775 -8.91928 0.121529

2.174 -8.81703 -8.93484 0.117814

2.177 -8.83632 -8.95047 0.11415

2.18 -8.85564 -8.96618 0.110535

2.183 -8.87499 -8.98196 0.10697

2.186 -8.89436 -8.99781 0.103452

2.189 -8.91375 -9.01374 0.0999825

2.192 -8.93317 -9.02973 0.0965598

2.195 -8.95262 -9.0458 0.0931836

2.198 -8.97209 -9.06194 0.0898532

2.201 -8.99158 -9.07815 0.0865681

2.204 -9.01109 -9.09442 0.0833276

2.207 -9.03064 -9.11077 0.0801312

2.21 -9.0502 -9.12718 0.0769782

2.213 -9.06979 -9.14366 0.0738682

2.216 -9.08941 -9.16021 0.0708005

2.219 -9.10905 -9.17682 0.0677746

2.222 -9.12871 -9.1935 0.0647899

2.225 -9.1484 -9.21024 0.061846

2.228 -9.16811 -9.22705 0.0589421

2.231 -9.18784 -9.24392 0.0560779

2.234 -9.2076 -9.26085 0.0532528

2.237 -9.22739 -9.27785 0.0504662

2.24 -9.24719 -9.29491 0.0477177

2.243 -9.26702 -9.31203 0.0450067

2.246 -9.28688 -9.32921 0.0423328

2.249 -9.30676 -9.34646 0.0396954

2.252 -9.32666 -9.36376 0.0370941

2.255 -9.34659 -9.38112 0.0345284

2.258 -9.36654 -9.39854 0.0319977

2.261 -9.38652 -9.41602 0.0295017

2.264 -9.40652 -9.43356 0.0270399

2.267 -9.42654 -9.45116 0.0246117

2.27 -9.44659 -9.46881 0.0222168

2.273 -9.46666 -9.48652 0.0198547

2.276 -9.48676 -9.50428 0.0175249

2.279 -9.50688 -9.5221 0.0152271

2.282 -9.52702 -9.53998 0.0129607

2.285 -9.54719 -9.55791 0.0107254

2.288 -9.56738 -9.5759 0.00852072

2.291 -9.58759 -9.59394 0.00634625

2.294 -9.60783 -9.61203 0.00420157

2.297 -9.62809 -9.63018 0.00208629

2.3 -9.64838 -9.64838 0

||e||=0.11826

**Код програми**

#include <math.h>

#include <iostream>

#include <fstream>

using namespace std;

double A = -1.150, B = -1.702, C = 0.662, D =-2.321, E = 2.136, alpha = 1, beta = 0,alpha1=-1,beta1=0.4;

double y(double x) {

return A\*x\*x + B\*x + C + 1/(D\*x + E);

};

double dy(double x) {

return 2\*A\*x + B - D/pow(D\*x + E, 2);

};

double ddy(double x) {

return 2\*A + 2\*D\*D/pow(D\*x + E, 3);

};

double p(double x) {

return 2\*x;

};

double q(double x) {

return -2;

};

double f(double x) {

return ddy(x) + p(x)\*dy(x) + q(x)\*y(x);

};

double f1 (double x) {

return alpha\*dy(x)+beta\*y(x);

}

double f2 (double x) {

return alpha1\*dy(x)+beta1\*y(x);

}

void method (int n, double \*d1, double \*d2, double \*d3, double \*F, double \*Y) {

double m;

for (int i = 1; i <= n; i++)

{

m = d1[i]/d2[i-1];

d2[i] = d2[i]-m\*d3[i-1];

F[i] = F[i]- m\*F[i-1];

}

Y[n] = F[n]/d2[n];

for (int i = n-1; i >= 0; i--)

Y[i] = (F[i]-d3[i]\*Y[i+1])/d2[i];

};

int main() {

const int n=100;

double a = 2, b = 2.3;

double h, g1, g2, eps;

double X[n+1], F[n+1], Y[n+1], e[n+1];

ofstream fout ("results.txt");

h = (double) fabs(b-a)/n;

g1 = f1(a);

g2 = f2(b);

eps = 0;

double arr[n+1][n+1];

for (int i = 0; i <= n; i++) {

X[i] = a+i\*h;

F[i] = f(X[i]);

if (i == 0) {

arr[i][i] = q(X[0])-2/(h\*h)+2\*h\*beta1\*(1/(h\*h)-p(X[0])/(2\*h))/alpha1;

arr[i][i+1] = 2/(h\*h);

F[i]=y(a);

}

else if (i==n) {

arr[i][i-1] = 0;

arr[i][i] = 1;

F[i] = y(b);

}

else{

arr[i][i-1] = 1/(h\*h)-p(X[i])/(2\*h);

arr[i][i] = -2/(h\*h)+q(X[i]);

arr[i][i+1] = 1/(h\*h)+p(X[i])/(2\*h);

}

}

double d1[n+1],d2[n+1],d3[n+1];

for (int i = 0; i <= n; i++) {

if (i==0)

d1[i] = 0;

else

d1[i] = arr[i][i-1];

if (i==n)

d3[i] = 0;

else

d3[i] = arr[i][i+1];

d2[i] = arr[i][i];

}

method (n, d1, d2, d3, F, Y);

cout << "h = "<< h << endl;

cout << "X[i]" << " " << "y(X[i])" << " " << "Y[i]" << " " << "e[i]" << endl;

for (int i = 0; i <= n; i++)

{

e[i] = y(X[i])-Y[i];

eps += pow (e[i],2);

fout << X[i] << " " << y(X[i]) << " " << Y[i] << " " << e[i] << endl;

cout << X[i] << " " << y(X[i]) << " " << Y[i] << " " << e[i] << endl;

}

eps = sqrt (eps\*h);

cout << "||e||=" << eps;

fout.close();

return 0;

}