**Отчет по курсовой работе №7** по курсу\_1\_\_\_\_\_

|  |  |
| --- | --- |
|  | Студент группы М80-105Б-21 Жилов Андрей, № по списку 3  Контакты www, e-mail: [klzxrcn3692@outlook.com](mailto:klzxrcn3692@outlook.com)\_\_\_\_\_\_\_\_\_\_\_  Работа выполнена: «24» апреля 2021г.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Преподаватель: В. К. Титов каф. 806\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отчет сдан «24» апреля 2021 г., итоговая оценка \_\_\_\_\_\_\_\_\_  Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

1. **Тема:** Разреженные матрицы
2. **Цель работы:** Составить программу на языке Си с процедурами и\или функциями для обработки прямоугольных разреженных матриц с элементами целого, вещественного, или комплексного типов.
3. **Задание:** (*Вариант №3*):

Вариант схемы размещения матрицы 3

Три вектора;

Вариант преобразования 3

Найти элемент матрицы, ближайший к заданному значению. Разделить на него элементы строки и столбца, на

пересечении которых он расположен. Если таких элементов несколько, обработать все.

1. **Оборудование(лабораторное)**:

ЭВМ -, процессор -, имя узла сети - с ОП - ГБ,

НМД - ГБ, терминал- адрес -, принтер –

Другие устройства –

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 3500U, с ОП 8 ГБ, НМД 256 ГБ. Монитор 1920x1080 пикс.

Другие устройства –

1. **Программное обеспечение:**

Операционная система семейства -, наименование - версия –

интерпретатор команд – версия

Система программирования - версия –

Редактор текстов - версия –

Утилиты операционной системы –

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства GNU/Linux, наименование Kali версия 5.10.0

интерпретатор команд Bash версия 5.1.8(1)-release

Система программирования \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

редактор текстов emacs версия 27.1

Утилиты операционной системы cmp, diff, tail, od, vi, vim, ed, head, du, xargs, indent, comm, grep, tee, sum, mc, awk, iconv, file,df,split, wc, join, tr,cut,tar,sed,patch,touch, paste, mktemp, dd,sort, uniq, nroff, gzip, bzip2, md5, find, cpp

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

1. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок,

На вход программе подается файл содержащий матрицу в сокращенном виде (из трех векторов). В файле также указаны количество строк, столбцов, ненулевых элементов, а также число, с которым мы будем сравнивать каждый ненулевой элемент матрицы и находить ближайший к нему, а затем делить на этот элемент строку и столбец в котором он находится.

После ввода всех данных, выводится матрица в сокращенном и полном виде, а затем начинается преобразование.

Для того чтобы избежать деления одного элемента два раза ( если в матрице два одинаковых элемента, максимально приближенных к вводимому числу, то мы должны будем разделить несколько строк и столбцов на данный элемент из-за чего получается деление одного элемента два враза) мы создаем копию полной матрицы и работаем только с неизмененными элементами. Проходимся по всей матрице, при нахождении максимально близкого к заданномузначению элемента делим на него строку и столбец.

После преобразования выводим полную и сокращенную матрицу вновь.

1. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Cursv2.c

#include <stdio.h>

float module(float k)

{

if(k<0)

return -k;

return k;

}

void short\_to\_full(float \*\*matrix, int n, int m, int k, int \*cip, int \*pi, float \*ye)

{

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

{

for (int j = 0; j < m; ++j)

{

matrix[i][j] = 0;

}

}

int p, l;

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

{

if (i == n - 1)

p = k + 1;

else

p = cip[i + 1];

if (p == 0)

{

p = cip[i + 2];

}

l = cip[i];

if (l != 0)

{

for (int j = l - 1; j < p - 1; ++j)

{

matrix[i][pi[j] - 1] = ye[j];

}

}

}

}

void full\_to\_short(float \*\*matrix, int n, int m, int \*cip, int \*pi, float \*ye)

{

int k = 0, zero;

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

{

cip[i] = k + 1;

zero = 0;

for (int j = 0; j < m; ++j)

{

if (matrix[i][j] != 0)

{

ye[k] = matrix[i][j];

pi[k] = j + 1;

k += 1;

zero += 1;

}

}

if (zero == 0)

cip[i] = 0;

}

}

void transform\_full\_matrix(int n, int m, float min\_diff, float \*\*matrix)

{

float epsilon=1.0;

while (1+epsilon>1)

epsilon/=2;

printf("epsilon is %f\n",epsilon);

float \*\*matrix\_modify;

matrix\_modify = new float\*[n];

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

matrix\_modify[i]=new float[m];

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

{

for(int j=0;j<m;++j)

matrix\_modify[i][j]=matrix[i][j];

}

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

{

for(int j=0;j<m;++j)

{

if((matrix\_modify[i][j]>=min\_diff-epsilon)&&(matrix\_modify[i][j]<=min\_diff+epsilon))

{

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

{

if(matrix\_modify[k][j]==matrix[k][j])

matrix[k][j]=matrix[k][j]/matrix\_modify[i][j];

}

for(int k=0;k<m;++k)

{

if(matrix\_modify[i][k]==matrix[i][k])

matrix[i][k]=matrix[i][k]/matrix\_modify[i][j];

}

}

}

}

delete[] matrix\_modify;

}

void print\_short\_matrix(int n,int k, int \*cip, int \*pi, float \*ye)

{

printf("CIP: ");

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

printf("%d ",cip[i]);

printf("\n");

printf("PI: ");

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

printf("%d ",pi[i]);

printf("\n");

printf("YE: ");

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

printf("%.1f ",ye[i]);

printf("\n");

}

void print\_full\_matrix(int n, int m, float \*\*matrix)

{

printf("Full matrix:\n");

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

{

for(int j=0;j<m;++j)

{

printf("%.1f ",matrix[i][j]);

}

printf("\n");

}

}

int main()

{

float \*\*matrix,\*ye;

int \*cip,\*pi;

int n,m,k;

float p,min\_diff,min\_element;

printf("n = \n");scanf("%d",&n);

printf("m = \n");scanf("%d",&m);

printf("k = \n");scanf("%d",&k);

printf("p = \n");scanf("%f",&p);

if(p==0)

{

printf("ERROR ERROR ERROR нельзя делить на 0!!!!!!!!!\n");

return 1;

}

matrix = new float\*[n];

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

matrix[i]=new float[m];

cip=new int[n];

pi=new int[k];

ye=new float[k];

printf("CIP: ");

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

scanf("%d",&cip[i]);

printf("\n");

printf("PI: ");

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

scanf("%d",&pi[i]);

printf("\n");

printf("YE: ");

min\_diff=p;

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

{

scanf("%f",&ye[i]);

if(module( p-ye[i])<min\_diff)

{

min\_element=ye[i];

min\_diff=p-ye[i];

}

}

printf("\n");

print\_short\_matrix(n, k, cip, pi, ye);

short\_to\_full(matrix, n, m,k, cip, pi, ye);

print\_full\_matrix(n, m, matrix);

transform\_full\_matrix(n, m, min\_element, matrix);

print\_full\_matrix(n, m, matrix);

full\_to\_short(matrix, n, m, cip, pi, ye);

print\_short\_matrix(n, k, cip, pi, ye);

delete[] cip;

delete[] pi;

delete[] ye;

delete[] matrix;

return 0;

}

*Пункты 1-7 отчета составляются сторого до начала лабораторной работы.*

*Допущен к выполнению работы.*  **Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Распечатка протокола**  (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

lain@navi:/mnt/d/mai/sem2/lab/curs7$ cat head\_curs\_7.txt

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| |

| Курсовая работа №7 |

| Разреженные матрицы |

| Студента группы М8О-105Б-21 |

| Жилова Андрея |

| |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

lain@navi:/mnt/d/mai/sem2/lab/curs7$ cat in.txt

7 5 29 4

1 4 8 12 16 20 25

1 4 5 1 2 3 5 1 3 4 5 1 2 3 5 1 3 4 5 1 2 3 4 5 1 2 3 4 5

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3.8 3 3 3 3 3 3 3

lain@navi:/mnt/d/mai/sem2/lab/curs7$ cat in1.txt

8 9 14 9

1 0 3 4 7 9 10 13

1 9 1 1 2 9 1 7 1 1 5 9 1 6

1 10 1 12 3 4 4 4 1 8 5 5 2 2

lain@navi:/mnt/d/mai/sem2/lab/curs7$ cat in2.txt

5 7 10 10

1 3 0 6 9

1 5 3 4 5 1 2 3 1 5

1 1 10 10 1 1 1 1 9 2

lain@navi:/mnt/d/mai/sem2/lab/curs7$ ./a.out < in.txt

n =

m =

k =

p =

CIP:

PI:

YE:

CIP: 1 4 8 12 16 20 25

PI: 1 4 5 1 2 3 5 1 3 4 5 1 2 3 5 1 3 4 5 1 2 3 4 5 1 2 3 4 5

YE: 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.8 3.0 3.0 3.0 3.0 3.0 3.0 3.0

Full matrix:

3.0 0.0 0.0 3.0 3.0

3.0 3.0 3.0 0.0 3.0

3.0 0.0 3.0 3.0 3.0

3.0 3.0 3.0 0.0 3.0

3.0 0.0 3.0 3.0 3.0

3.0 3.0 3.8 3.0 3.0

3.0 3.0 3.0 3.0 3.0

epsilon is 0.000000

Full matrix:

3.0 0.0 0.0 3.0 3.0

3.0 3.0 0.8 0.0 3.0

3.0 0.0 0.8 3.0 3.0

3.0 3.0 0.8 0.0 3.0

3.0 0.0 0.8 3.0 3.0

0.8 0.8 1.0 0.8 0.8

3.0 3.0 0.8 3.0 3.0

CIP: 1 4 8 12 16 20 25

PI: 1 4 5 1 2 3 5 1 3 4 5 1 2 3 5 1 3 4 5 1 2 3 4 5 1 2 3 4 5

YE: 3.0 3.0 3.0 3.0 3.0 0.8 3.0 3.0 0.8 3.0 3.0 3.0 3.0 0.8 3.0 3.0 0.8 3.0 3.0 0.8 0.8 1.0 0.8 0.8 3.0 3.0 0.8 3.0 3.0

lain@navi:/mnt/d/mai/sem2/lab/curs7$ ./a.out < in1.txt

n =

m =

k =

p =

CIP:

PI:

YE:

CIP: 1 0 3 4 7 9 10 13

PI: 1 9 1 1 2 9 1 7 1 1 5 9 1 6

YE: 1.0 10.0 1.0 12.0 3.0 4.0 4.0 4.0 1.0 8.0 5.0 5.0 2.0 2.0

Full matrix:

1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 10.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

12.0 3.0 0.0 0.0 0.0 0.0 0.0 0.0 4.0

4.0 0.0 0.0 0.0 0.0 0.0 4.0 0.0 0.0

1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

8.0 0.0 0.0 0.0 5.0 0.0 0.0 0.0 5.0

2.0 0.0 0.0 0.0 0.0 2.0 0.0 0.0 0.0

epsilon is 0.000000

Full matrix:

0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

12.0 3.0 0.0 0.0 0.0 0.0 0.0 0.0 0.4

4.0 0.0 0.0 0.0 0.0 0.0 4.0 0.0 0.0

1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

8.0 0.0 0.0 0.0 5.0 0.0 0.0 0.0 0.5

2.0 0.0 0.0 0.0 0.0 2.0 0.0 0.0 0.0

CIP: 1 0 3 4 7 9 10 13

PI: 1 9 1 1 2 9 1 7 1 1 5 9 1 6

YE: 0.1 1.0 1.0 12.0 3.0 0.4 4.0 4.0 1.0 8.0 5.0 0.5 2.0 2.0

lain@navi:/mnt/d/mai/sem2/lab/curs7$ ./a.out < in2.txt

n =

m =

k =

p =

CIP:

PI:

YE:

CIP: 1 3 0 6 9

PI: 1 5 3 4 5 1 2 3 1 5

YE: 1.0 1.0 10.0 10.0 1.0 1.0 1.0 1.0 9.0 2.0

Full matrix:

1.0 0.0 0.0 0.0 1.0 0.0 0.0

0.0 0.0 10.0 10.0 1.0 0.0 0.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0

1.0 1.0 1.0 0.0 0.0 0.0 0.0

9.0 0.0 0.0 0.0 2.0 0.0 0.0

epsilon is 0.000000

Full matrix:

1.0 0.0 0.0 0.0 1.0 0.0 0.0

0.0 0.0 1.0 1.0 0.1 0.0 0.0

0.0 0.0 0.0 0.0 0.0 0.0 0.0

1.0 1.0 0.1 0.0 0.0 0.0 0.0

9.0 0.0 0.0 0.0 2.0 0.0 0.0

CIP: 1 3 0 6 9

PI: 1 5 3 4 5 1 2 3 1 5

YE: 1.0 1.0 1.0 1.0 0.1 1.0 1.0 0.1 9.0 2.0

1. **Дневник отладки** должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  |  |  |  |  |  |  |

1. **Замечания автора по существу работы:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Выводы**:

В результате выполнения данной лабораторной работы я научился работать с разреженными матрицами на языке Си

Недочеты при выполнении задания могут быть устранены следующим образом:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_