

Лабораторная работа №9

- 1) Тема: Препроцессор, модульная сборка программы
- 2) Напишите программу из нескольких файлов (модулей), включая файл основной программы. Файлы должны содержать вынесенные отдельно функции для выделения памяти под динамические двумерные и одномерные массивы и функции для перемножения матриц. Собрать проект используя утилиту Make.
- 3)

CC=gcc

CFLAGS=-std=c17

DEPENDENCIES=lab.h

OBJECT_DEPENDENCIES=Lab_9.o Lab_9_Func.o

LIBRARIES=-lm

.DEFAULT_GOAL=program

%.o: %.c \$(DEPENDENCIES)

\$(CC) \$(CFLAGS) -c -o \$@ \$<

program:\$(OBJECT_DEPENDENCIES)

\$(CC) \$(CFLAGS) -o \$@ \$^ \$(LIBRARIES)

- 4)

Имя	Смысл	Тип
createmat	Функция, создающая двумерный массив	double** createmat (int row, int coll)
createmas	Функция, создающая одномерный массив	double* createmas(int sized)
proizmas	Функция произведения одномерного и двумерного массивов	double* proizmas(double**A, double*B, int n,int k)
proizmat	Функция произведения матриц	double** proizmat(double**A, double **B, int n,int k)
row	Промежуточная переменная функции (кол-во строк)	Int

coll	Промежуточная переменная функции (кол-во столбцов)	Int
NewMt	Результат функции (созданная матрица)	double
i	Индекс элемента массива	Int
*NewMs	Результат функции (созданный массив)	double *
sized	Промежуточная переменная функции (размер массива)	Int
A	Промежуточная переменная функции (матрица)	Double
B	Промежуточная переменная функции (массив)	Double
n	Промежуточная переменная функции (кол-во строк)	Int
k	Промежуточная переменная функции (кол-во столбцов)	Int
j	Индекс элемента массива	Int
l	Индекс элемента массива	Int
B	Промежуточная переменная функции (матрица)	Double
C	Результат функции (Произведения матриц)	Double
C	Результат функции (Произведения матрицы и массива)	Double
R	Количество строк	Int
C	Количество столбцов	Int
X	Матрица 1	Double
Y	Матрица 2	Double
Z	Массив	Double
Q	Результат произведения(Матрица)	Double
W	Результат произведения(Массив)	Double

5) Lab_9_Func.c

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <locale.h>
```

```
#include "lab.h"
```

```
double** createmat (int row, int coll)
```

```
{
```

```
double** NewMt=(double **)malloc(row*sizeof(double*));
```

```
for (int i=0;i<row;i++)
```

```
{
```

```
    NewMt[i] = (double *) malloc(coll*sizeof(double));
```

```
}
```

```
return (NewMt);
```

```
}
```

```
double* createmas(int sized)
```

```
{
```

```
    double *NewMs=(double *) malloc(sized*sizeof(double));
```

```
    return(NewMs);
```

```
}
```

```
double* proizmas(double**A, double*B, int n,int k){
```

```
double *C;
```

```
C=(double *) malloc(n*sizeof(double));
```

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

```
{
```

```
    for (int j=0;j<k;j++)
```

```
    {
```

```
        C[i]+=A[i][j]*B[j];
```

```
    }
```

```
}
```

```
return(C);
```

```
}
```

```
double** proizmat(double**A, double **B, int n,int k){
```

```
double **C;
C=(double **) malloc (n * sizeof(double *));
    for (int i=0;i<n;i++)
    {
        C[i]=(double *) malloc(k*sizeof(double));
    }
```

```
for(int i = 0; i < n; ++i) {
    for(int j = 0; j < k; ++j) {
        C[i][j] = 0;
        for(int l = 0; l < n; ++l)
            C[i][j] += A[i][l] * B[l][j];
    }
}
return(C);
}
```

lab.h

```
#define LAB_H
```

```
double** createmat (int row, int coll);
double* createmas(int sized);
double* proizmas(double**A, double*B, int n,int k);
double** proizmat(double**A, double **B, int n,int k);
```

```
#include <stdio.h>
#include <stdlib.h>
#include <locale.h>
#include "lab.h"

int main()
{
    char *locale=setlocale(LC_ALL, "");
    int R,C;
    printf("Enter number of rows and collumns\n");
    scanf("%d",&R);
    scanf("%d",&C);
    double **X = createmat(R,C);
    double **Y = createmat(R,C);
    double *Z = createmas(R);
    printf("Enter matrix 1\n");
    for (int i=0;i<R;i++)
    {
        for (int j=0;j<C;j++)
        {
            scanf("%lf",&X[i][j]);
        }
    }
    printf("Enter matrix 2\n");
    for (int i=0;i<R;i++)
    {
```

```

    for (int j=0;j<C;j++)
    {
        scanf("%lf",&Y[i][j]);
    }
}
printf("Enter massiv\n");
for (int i=0; i<R; i++)
{
    scanf("%lf",&Z[i]);
}
printf("Matrix 1*2 = \n");
double **Q;
Q=proizmat(X,Y,R,C);
for (int i=0;i<R;i++)
{
    for (int j=0;j<C;j++)
    {
        printf("%lf\n",Q[i][j]);
    }
}
printf("Matrix 1 * Massiv = \n");
double *W;
W=proizmas(X,Z,R,C);
for (int i=0; i<R; i++)
{
    printf("%lf\n",W[i]);
}
scanf("%d");
return 0;

```

}

Makefile

CC=gcc

CFLAGS=-std=c17

DEPENDENCIES=lab.h

OBJECT_DEPENDENCIES=Lab_9.o Lab_9_Func.o

LIBRARIES=-lm

.DEFAULT_GOAL=program

%.o: %.c \$(DEPENDENCIES)

\$(CC) \$(CFLAGS) -c -o \$@ \$<

program:\$(OBJECT_DEPENDENCIES)

\$(CC) \$(CFLAGS) -o \$@ \$^ \$(LIBRARIES)

6)

```
Enter number of rows and collumns
2
2
Enter matrix 1
1
2
3
4
Enter matrix 2
5
6
7
8
Enter massiv
1
2
Matrix 1*2 =
19,000000
22,000000
43,000000
50,000000
Matrix 1 * Massiv =
3,000000
14,000000
```