

Лабораторная работа №2. Разработка подпрограмм.

Часть 1. Разработка подпрограмм с параметром-массивом

Целью данной работы является получение навыков создания подпрограмм. В задачах этого раздела требуется логически независимые или повторяющиеся последовательности действий оформить в виде отдельных подпрограмм, обращение к которым происходит из основной программы. При разработке программы составить алгоритм по отдельности, как для функции, так и для основной части программы, использующей функцию. Все размерности массивов задаются с клавиатуры.

Номер решаемой задачи соответствует номеру варианта, выданному преподавателем.

Задание на программирование

1. Создать функцию, вычисляющую число перемен знака в массиве X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_7 и B_1, B_2, \dots, B_9 .
2. Создать функцию, вычисляющую количество элементов среди X_1, X_2, \dots, X_n , значения которых совпадают со значениями элементов массива Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_7 ; B_1, B_2, \dots, B_7 и C_1, C_2, \dots, C_5 ; D_1, D_2, \dots, D_5 .
3. Создать функцию, вычисляющую сумму отрицательных элементов массива Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_7 и B_1, B_2, \dots, B_6 .
4. Создать функцию, вычисляющую произведение положительных элементов среди элементов массива X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_6 и B_1, B_2, \dots, B_8 .
5. Создать функцию, вычисляющую полусумму минимального и максимального элементов массива X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_6 и B_1, B_2, \dots, B_7 .
6. Создать функцию, вычисляющую значение многочлена следующего вида $Y_1 \cdot Z^{n-1} + Y_2 \cdot Z^{n-2} + \dots + Y_{n-1} \cdot Z + Y_n$ (Y – массив коэффициентов, Z – аргумент). Использовать функцию для вычисления значений двух многочленов, с массивами коэффициентов A_1, A_2, \dots, A_5 и B_1, B_2, \dots, B_7 и аргументами P и T , соответственно.

7. Создать функцию, вычисляющую количество нулей в массиве X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_6 и B_1, B_2, \dots, B_7 .
8. Создать функцию, вычисляющую наибольшую абсолютную величину элемента среди элементов массива Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_7 и B_1, B_2, \dots, B_9 .
9. Создать функцию, вычисляющую число элементов массива Y_1, Y_2, \dots, Y_n , значения которых совпадают со значениями элементов X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_5 ; B_1, B_2, \dots, B_5 и C_1, C_2, \dots, C_4 ; D_1, D_2, \dots, D_4 .
10. Создать функцию, вычисляющую скалярное произведение элементов массивов X_1, X_2, \dots, X_n и Y_1, Y_2, \dots, Y_n , равное $\sum_{i=1}^n X_i Y_i$. Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_6 ; B_1, B_2, \dots, B_6 и C_1, C_2, \dots, C_5 ; D_1, D_2, \dots, D_5 .
11. Создать функцию, вычисляющую произведение максимальных элементов в массивах X_1, X_2, \dots, X_n и Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_6 ; B_1, B_2, \dots, B_6 и C_1, C_2, \dots, C_5 ; D_1, D_2, \dots, D_5 .
12. Создать функцию, вычисляющую число элементов массива X_1, X_2, \dots, X_n , которые больше максимального элемента в массиве Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_5 ; B_1, B_2, \dots, B_5 и C_1, C_2, \dots, C_8 ; D_1, D_2, \dots, D_8 .
13. Создать функцию, вычисляющую число элементов среди X_1, X_2, \dots, X_n , которые не превосходят максимального элемента Y_1, Y_2, \dots, Y_n и в то же время не меньше его минимального элемента. Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_7 ; B_1, B_2, \dots, B_7 и C_1, C_2, \dots, C_8 ; D_1, D_2, \dots, D_8 .
14. Создать функцию, вычисляющую число элементов массива X_1, X_2, \dots, X_n , которые делятся на 7 без остатка (обоснованно выбрать тип элементов массива X). Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_6 и B_1, B_2, \dots, B_7 .

15. Создать функцию, вычисляющую расстояние от начала координат до точки n -мерного пространства с координатами, задаваемые массивом X_1, X_2, \dots, X_n (оно равно корню квадратному из $\sum_{i=1}^n X_i^2$). Использовать функцию для вычисления указанных расстояний для двух точек с координатами, задаваемыми массивами A_1, A_2, \dots, A_5 и B_1, B_2, \dots, B_7 .
16. Создать функцию, вычисляющую значение наибольшего элемента главной диагонали матрицы A , состоящей из n строк и n столбцов. Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице T из 4 строк и 4 столбцов и матрице P из 5 строк и 5 столбцов.
17. Создать функцию, вычисляющую количество положительных элементов в двух заданных строках матрицы A , состоящей из m строк и n столбцов. Использовать функцию для вычисления указанных величин в 1 и 4 строках матрицы T из 4 строк и 6 столбцов и в 2 и 3 строках матрицы P из 3 строк и 5 столбцов.
18. Создать функцию, вычисляющую абсолютную величину разности максимальных элементов двух заданных столбцов матрицы A , состоящей из m строк и n столбцов. Использовать функцию для вычисления указанных величин в 1 и 3 столбцах матрицы D из 4 строк и 5 столбцов и в 2 и 4 столбцах матрицы E из 5 строк и 4 столбцов.
19. Создать функцию, вычисляющую общее количество нулей в i -й и последней строке, i -м и последнем столбце матрицы A , состоящей из n строк и n столбцов. Использовать функцию для вычисления указанных величин в матрице D из 4 строк и 4 столбцов (для $i=2$) и в матрицы E из 5 строк и 5 столбцов (для $i=1$).
20. Создать функцию, вычисляющую количество локальных минимумов матрицы A , состоящей из m строк и n столбцов (локальный минимум – это значение элемента, который меньше двух соседних элементов слева и справа). Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице T из 3 строк и 4 столбцов и матрице P из 2 строк и 5 столбцов.
21. Создать функцию, вычисляющую среднее арифметическое элементов над главной диагональю матрицы A , состоящей из n строк и n столбцов. Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице R из 4 строк и 4 столбцов и матрице P из 5 строк и 5 столбцов.

22. Создать функцию, вычисляющую количество строк матрицы A , состоящей из m строк и n столбцов, сумма элементов каждой из которых меньше нуля. Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице C из 4 строк и 5 столбцов и матрице D из 5 строк и 3 столбцов.
23. Создать функцию, вычисляющую максимальный элемент в заданной группе соседних строк матрицы A , состоящей из m строк и n столбцов. Использовать функцию для вычисления указанных величин в матрице D из 4 строк и 5 столбцов (для строк с 1 по 3) и в матрицы E из 3 строк и 5 столбцов (для строк с 2 по 3).
24. Создать функцию, вычисляющую наибольшее число подряд идущих положительных элементов среди элементов массива X_1, X_2, \dots, X_n . Использовать функцию для вычисления указанных величин в следующих двух массивах A_1, A_2, \dots, A_7 и B_1, B_2, \dots, B_9 .
25. Создать функцию, вычисляющую наименьший элемент в совокупности элементов двух массивов X_1, X_2, \dots, X_n ; Y_1, Y_2, \dots, Y_n . Использовать функцию для вычисления указанных величин для следующих двух пар массивов A_1, A_2, \dots, A_5 ; B_1, B_2, \dots, B_5 и C_1, C_2, \dots, C_6 ; D_1, D_2, \dots, D_6 .
26. Создать функцию, вычисляющую разность сумм элементов над и под главной диагональю матрицы A , состоящей из n строк и n столбцов. Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице Z из 4 строк и 4 столбцов и матрице W из 5 строк и 5 столбцов.
27. Создать функцию, вычисляющую общее количество отрицательных элементов на главной диагонали и на двух соседних с ней (сверху и снизу) диагоналях матрицы A , состоящей из n строк и n столбцов. Использовать функцию для вычисления указанных величин в следующих двух матрицах: матрице B из 5 строк и 5 столбцов и матрице C из 4 строк и 4 столбцов.
28. Создать функцию, вычисляющую наименьшую сумму строки в матрице A , состоящей из m строк и n столбцов. Использовать функцию для вычисления указанных величин в матрице P из 5 строк и 3 столбцов и в матрицы T из 4 строк и 4 столбцов.
29. Создать функцию, вычисляющую наибольший из минимальных элементов строк матрицы A , состоящей из m строк и n столбцов. Использовать функцию для вычисления указанных величин в матрице D из 4 строк и 5 столбцов и в матрицы E из 3 строк и 5 столбцов.

Часть 2. Многомодульные программы

Задачи направлены на получение навыков в разработке многомодульных программ с выделением подпрограмм. В задании необходимо разместить все функции в отдельном модуле. Размеры всех структур динамические и задаются пользователем. Постараться реализовать оконный интерфейс.

Номер решаемой задачи соответствует номеру варианта, выданному преподавателем.

Пример решения задачи

В прямоугольной матрице в каждом столбце поставить на первое место максимальный элемент столбца и, если среди полученных элементов первой строки не окажется элементов, по модулю меньших заданной величины, разделить элементы последней строки на соответствующие элементы первой строки.

Проанализировав задание, можно выделить шесть подпрограмм:

- нахождение номера максимального элемента в произвольном столбце;
- перестановка максимальных элементов в первую строку матрицы;
- проверка на существование в первой строке элементов по модулю меньших заданной величины;
- деление элементов последней строки на соответствующие элементы первой строки;
- ввод матрицы с клавиатуры;
- вывод матрицы на экран монитора.

Рассмотрим вариант решения в стиле языка Си.

```
/*modul.h*/  
#include<stdio.h>  
#include<conio.h>  
#include<math.h>  
#include<stdlib.h>  
void vvod(float **, int, int);  
void vivod(float **, int, int);  
int maxelem(float **, int, int);  
void perest(float **, int, int);  
int prover(float **, int, float);  
void delenie(float **, int, int);  
  
void vvod(float **matr, int n, int m)
```

```

{
    int i, j;
    for(i=0; i<n; i++)
        for(j=0; j<m; j++)
        {
            printf("\nelem[%d,%d]=", i+1, j+1);
            scanf("%f", &matr[i][j]);
        }
}

void vivod(float **matr, int n, int m)
{
    int i, j;
    printf("\n Matriză\n ");
    for(i=0; i<n; i++)
        for(j=0; j<m; j++)
            printf("%2.2f\t ", matr[i][j]);
    printf("\n");
}

int maxelem(float **matr, int n, int k)
{
    int i, max=0;
    for(i=1; i<n; i++)
        if (matr[max][k] < matr[i][k])
            max=i;
    return max;
}

void perest(float **matr, int n, int m)
{
    int i, j;
    float d;
    for(i=0; i<m; i++)
    {
        j=maxelem(matr, n, i);
        d=matr[j][i];
        matr[j][i]=matr[0][i];
        matr[0][i]=d;
    }
}

int prover(float **matr, int m, float v)
{
    int i, j=-1;
    for(i=0; i<m; i++)
        if(fabs(matr[1][i])>= v)
            j=0;
    return j;
}

void delenie(float **matr, int n, int m)
{
    int i;
    for(i=0; i<m; i++)
        matr[n-1][i]=matr[n-1][i]/matr[0][i];
}

```

```

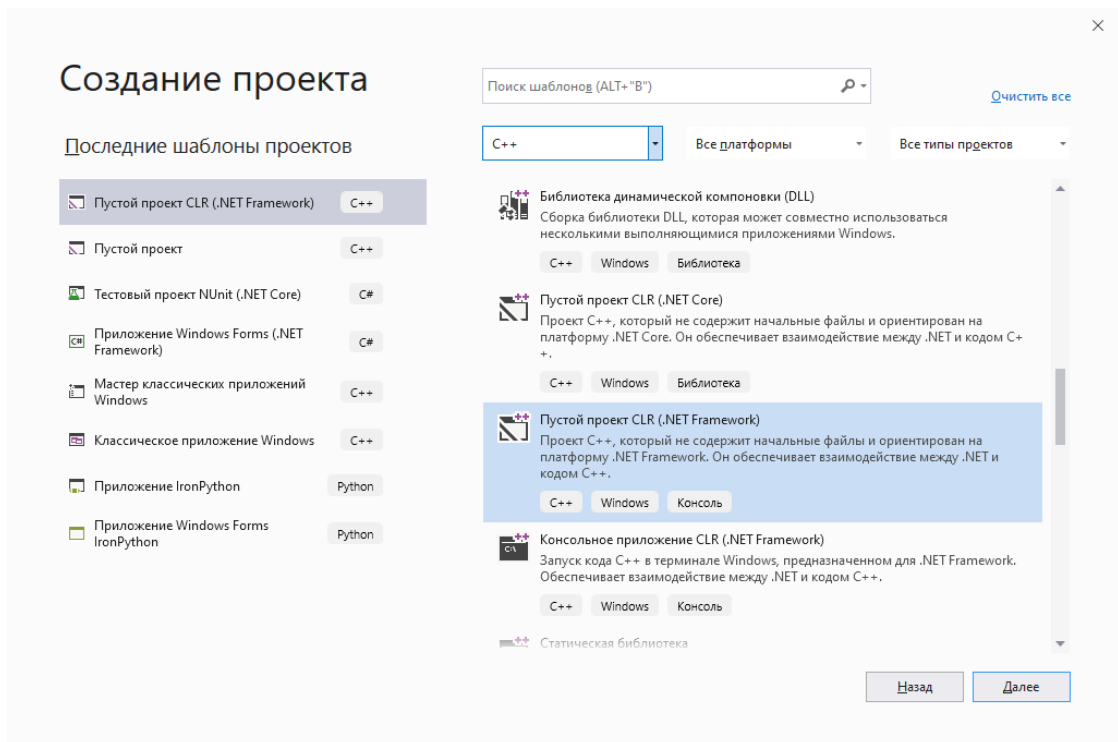
}

#include "modul.h"
void main()
{
    int n, m, i;
    float **a, vel;
    puts("Vvedite kol str n matritzi A: ");
    scanf("%d",&n);
    puts("Vvedite kol stol m matritzi A: ");
    scanf("%d",&m);
    a=(float**)malloc(n*sizeof(float*));
    for(i=0; i<n; i++)
        a[i]=(float*)malloc(m*sizeof(float));
    vvod(a,n,m);
    puts("Vvedite velichiny vel: ");
    scanf("%f",&vel);
    perest(a,n,m);
    if (prover(a,m,vel)==0)
        puts("V I-str est zadan element");
    else
    {
        delenie(a,n,m);
        vivod(a,n,m);
    }
    free(a);
}

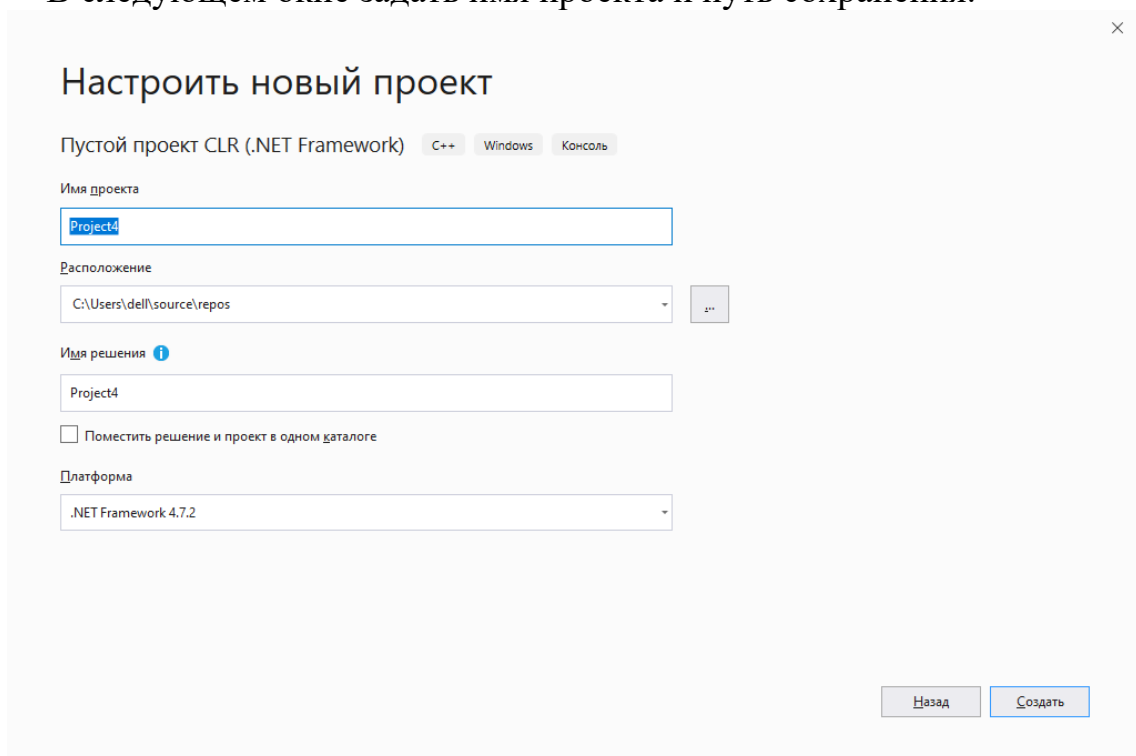
```

Работа с динамическими данными реализовать в стиле C++!!!

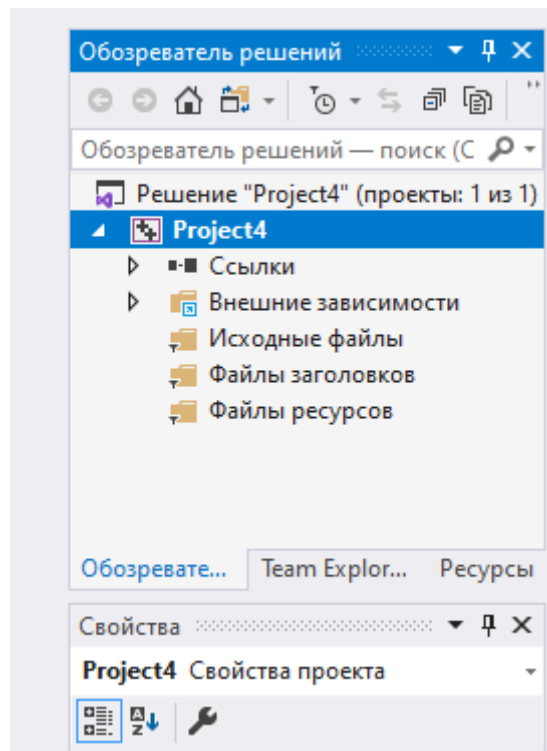
Для создания оконного интерфейса среди предложенных шаблонов найти **Пустой проект CLR(.NET Framework)** и нажать **Далее**.



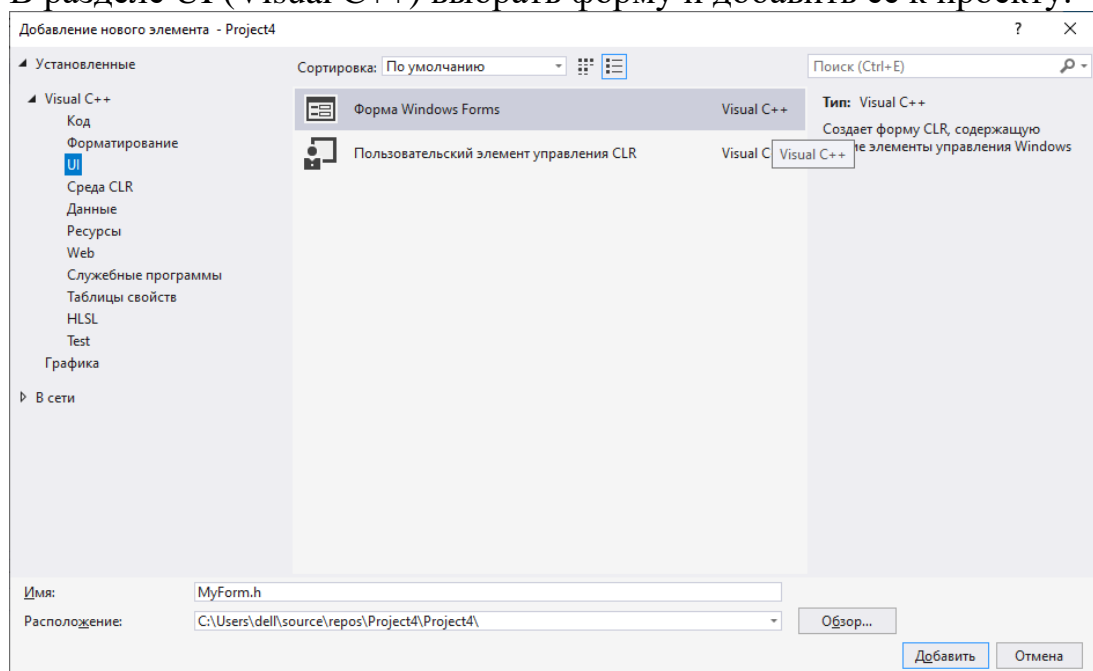
В следующем окне задать имя проекта и путь сохранения.



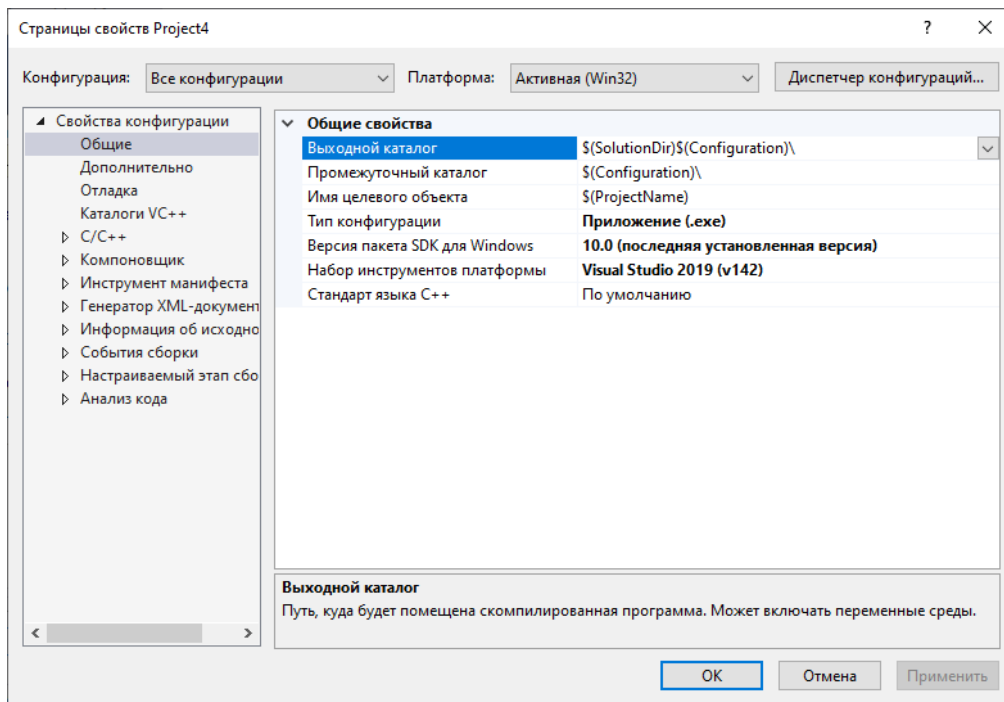
В созданный пустой проект необходимо добавить Форму (Form). В выпадающем меню правой кнопки на имени проекта в **Обозревателе решений** выбрать **Добавить->Создать элемент**.



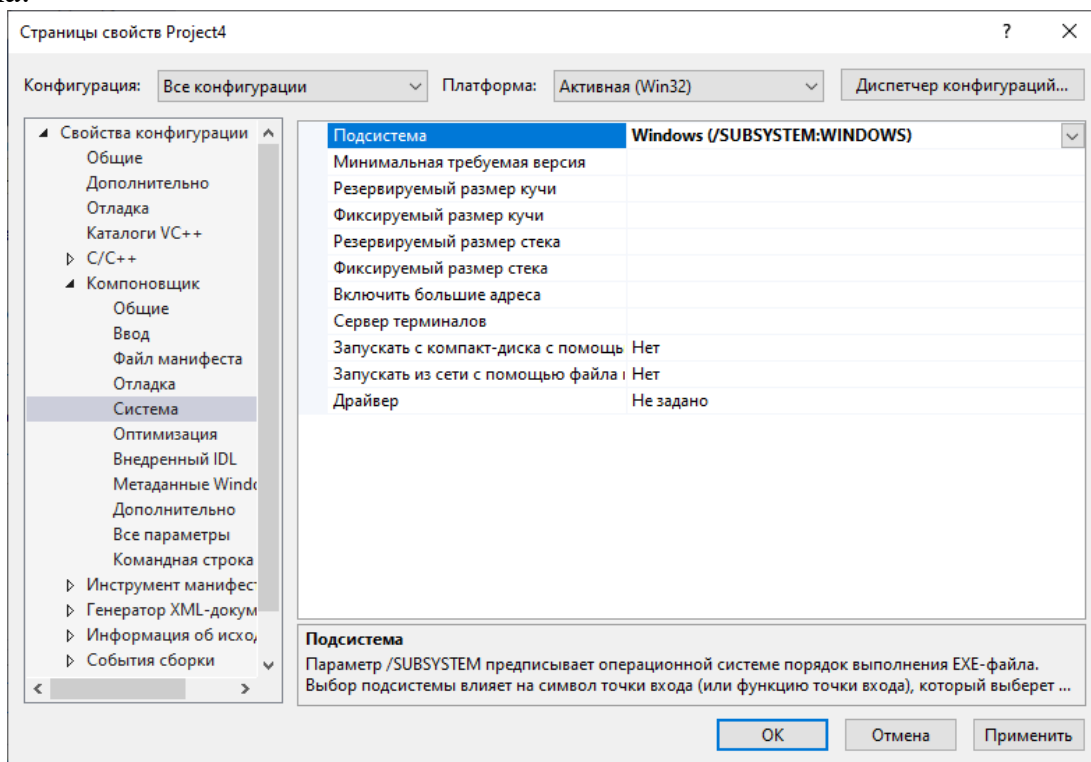
В разделе UI (Visual C++) выбрать форму и добавить ее к проекту.



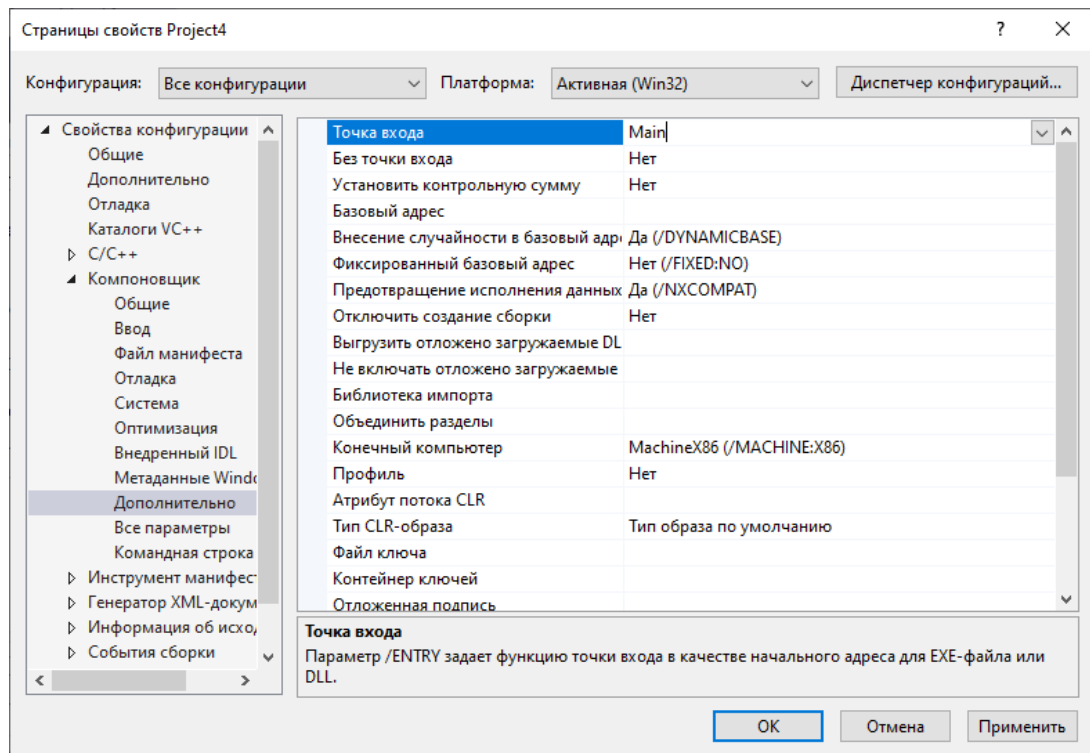
Далее, через обозреватель решений перейти в свойства проекта.



Для компоновщика в строке **Подсистема** (раздела **Система**) установить **Windows (/SUBSYSTEM:WINDOWS)**, выбрав из выпадающего списка.



А также в строке **Точка входа** (раздела **Дополнительно**) указать функцию **Main**.



Применить и нажать Ок. Затем через обозреватель открыть файл MyForm.cpp (или как вы его назвали при добавлении формы) и добавить в него следующий код, не забыв правильно указать имя текущего проекта:

```
using namespace System;
using namespace System;
using namespace System::Windows::Forms;

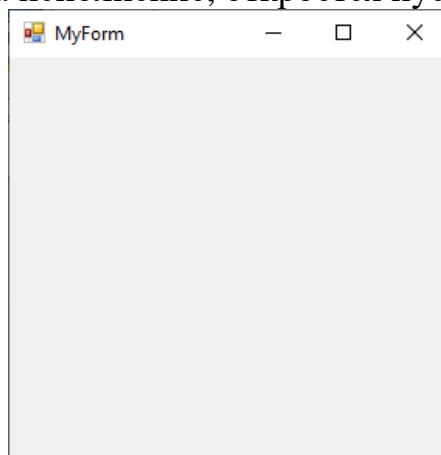
[STAThreadAttribute]

void Main(array< String>^ args) {

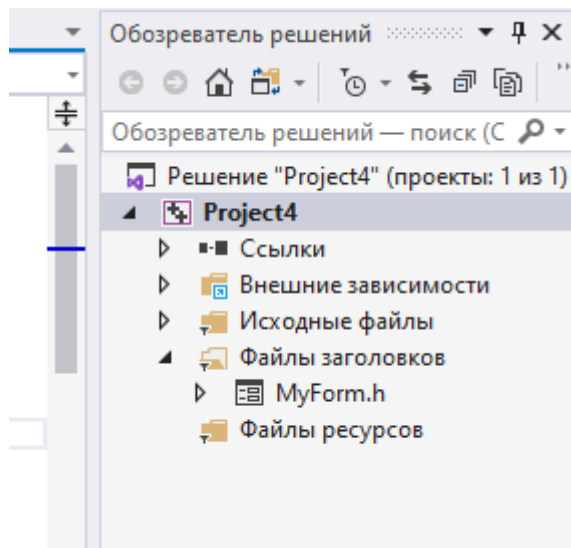
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);
    Project4::MyForm form; //указать имя своего проекта
    Application::Run(% form);

}
```

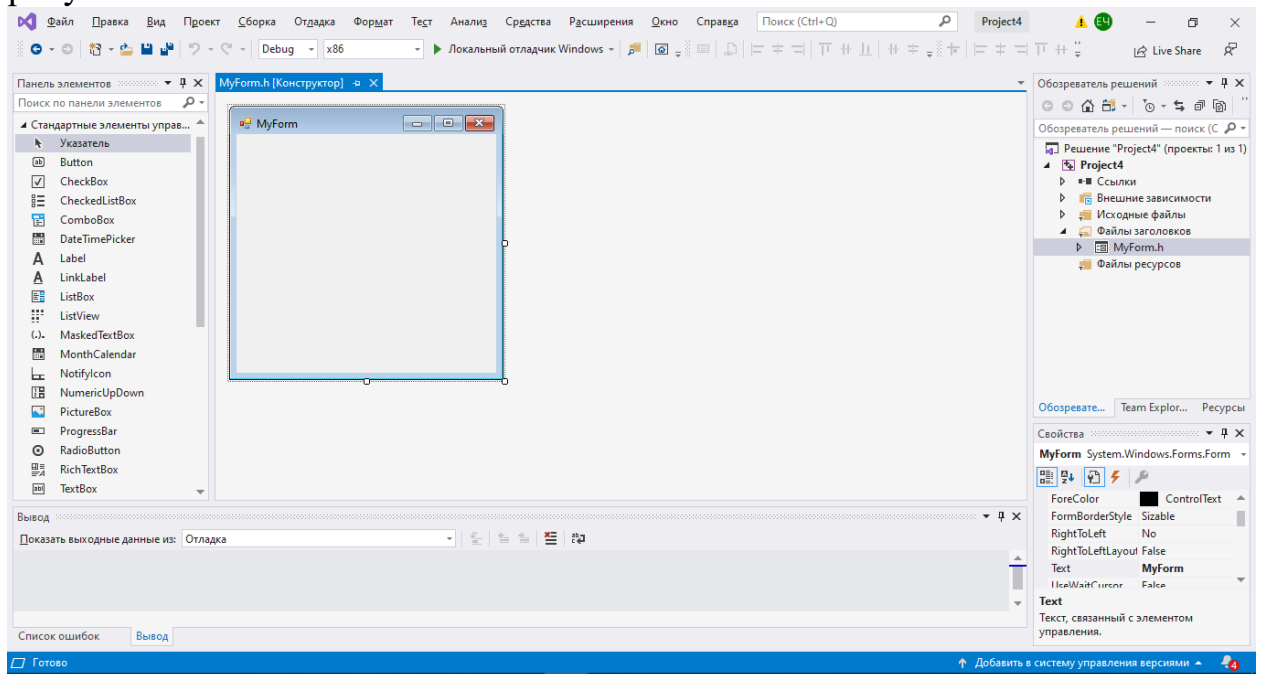
Запустив проект на исполнение, откроется пустая форма.



Чтобы все изменения вступили в силу надо закрыть и открыть проект заново. После открытия можно открыть конструктор формы через обозреватель (кликнув на файл MyForm.h)



Открыть панель доступных элементов можно открыть через меню **Вид** (Ctrl+Alt+X) и закрепить ее в удобном для работы месте, например как рисунке ниже.



Задание на программирование

1. Если первая строка прямоугольной матрицы имеет максимальное количество отрицательных элементов, проверить, как изменится среднее арифметическое всей матрицы, если заменить все отрицательные элементы матрицы их модулями.
2. Если в прямоугольной матрице все суммы элементов строк попадают на заданный отрезок, определить номер строки с максимальной суммой элементов, иначе определить номера строк, сумма элементов которых не попала на заданный отрезок.
3. Определить столбец прямоугольной матрицы с максимальной суммой элементов и, если его номер больше заданного, сформировать матрицу из столбцов исходной до найденного столбца, иначе сформировать массив из элементов заданного столбца.
4. Если заданная квадратная целочисленная матрица является треугольной (элементы выше главной диагонали равны нулю), вычислить ее среднее арифметическое, иначе определить, сколько элементов, лежащих выше главной диагонали, отличны от нуля.
5. Если k -й столбец прямоугольной матрицы имеет минимальную сумму элементов, определить сумму элементов столбцов до k -го, иначе сумму элементов столбцов после k -го.
6. Если целочисленная квадратная матрица симметрична относительно главной диагонали, обнулить все элементы, лежащие выше главной диагонали, и определить сумму элементов, лежащих ниже главной диагонали.
7. Переставить в каждом столбце прямоугольной матрицы все отрицательные элементы в конец столбца. Распечатать часть полученной матрицы, состоящую из n первых строк, не имеющих отрицательных элементов.
8. Если все точки плоскости, заданные своими координатами, попадают в круг с радиусом R и центром в начале координат, определить их среднюю абсциссу и ординату, иначе распечатать номера точек, не попавших в заданный круг.
9. Если столбцы заданной прямоугольной целочисленной матрицы расположены в порядке возрастания числа нулевых элементов в них, то подсчитать число нулевых элементов во всей матрице, иначе определить столбец с максимальным количеством нулей.
10. Если максимальный элемент квадратной матрицы находится выше главной диагонали, транспонировать матрицу, иначе определить сумму элементов строки и столбца с номерами, равными индексам максимального элемента.
11. Дана квадратная матрица. Увеличить все элементы строки с минимальной суммой элементов на среднее арифметическое элементов матрицы, лежащих выше главной диагонали.

12. Изменить заданную прямоугольную матрицу так, чтобы на первом месте стояла строка с максимальной, а на последнем месте строка с минимальной суммой элементов, сохранив все элементы исходной матрицы.
13. $Y = \frac{1}{\sqrt{x}}$ можно вычислить по итерационной формуле: $Y_{i+1} = \frac{3}{2}Y_i - \frac{1}{2}xY_i^3$ с точностью E и $Y_0 = \frac{1}{x}$. Вычислить $Z = \frac{1}{\sqrt{A_1}} + \frac{1}{\sqrt{A_2}} + \dots + \frac{1}{\sqrt{A_n}}$ для заданного массива A и заданной точности EPS .
14. Если все заданные точки плоскости принадлежат первой четверти, определить координаты точки, наиболее удаленной от начала координат, иначе определить координаты точек, не попавших в первую четверть.
15. Найти максимальный среди отрицательных элементов прямоугольной матрицы и минимальный среди положительных. Если они отличаются по модулю меньше, чем на заданную величину, заменить все отрицательные элементы матрицы их модулями.
16. Определить по экзаменационной ведомости, попадает ли группа на конкурс лучших групп. Условия конкурса: средний балл группы выше четырех, отсутствие неуспевающих, число студентов, не имеющих троек, больше половины всех студентов группы.
17. Подсчитать как изменится среднее арифметическое элементов матрицы, если во всех столбцах с номерами, большими, чем номер столбца с максимальным количеством отрицательных элементов, заменить все отрицательные элементы их модулями.
18. В заданной прямоугольной матрице поставить на первое место столбец с наименьшим количеством нулевых элементов, переставив все нули в конец этого столбца. (Измененная матрица должна содержать все элементы исходной матрицы).
19. В заданной прямоугольной матрице определить столбец, в котором все элементы расположены в порядке возрастания их значений. Если такого столбца нет, поменять порядок следования столбцов на противоположный.
20. Дана квадратная матрица. Если номер столбца с максимальной суммой элементов совпадает с номером строки с максимальной суммой элементов, определить сумму найденных элементов строки и столбца, иначе распечатать номера найденных строки и столбца с максимальными суммами.
21. Даны два целочисленных массива с положительными элементами: X_1, X_2, \dots, X_n и Y_1, Y_2, \dots, Y_n . Если все элементы массива X меньше всех элементов массива Y с соответствующими индексами, сформировать массив Z_1, Z_2, \dots, Z_n по правилу: $Z_i = \frac{Y_i!}{X_i!(Y_i + X_i)!}$.

22. Из целочисленной прямоугольной матрицы исключить столбец с максимальным числом нулевых элементов, сохранив все остальные элементы матрицы в том же порядке.
23. Если в первом столбце прямоугольной матрицы все элементы по модулю больше заданной величины, разделить столбец с максимальным средним арифметическим элементов на соответствующие элементы первого столбца.
24. Если все элементы главной диагонали квадратной целочисленной матрицы упорядочены по убыванию значений, заменить нулями все отрицательные элементы, лежащие выше главной диагонали, определив, как изменится при этом сумма элементов всей матрицы.
25. Если максимальный и минимальный элемент прямоугольной матрицы не попадают на заданный отрезок, определить, как изменится сумма элементов матрицы, в случае замены всех элементов, выходящих за границы заданного отрезка соответствующими границами.
26. В каждом столбце прямоугольной матрицы поставить на первое место максимальный по модулю элемент столбца, определив после этого среднее арифметическое элементов первой строки. (Измененная матрица должна содержать все элементы исходной матрицы).
27. Известно, что корень n -й степени из X может быть вычислен по итерационной формуле: $Y_i = Y_{i-1} + \left(\frac{x}{Y_{i-1}^{n-1}} - Y_{i-1} \right) \frac{1}{n}$ точно с ϵ и при $Y_0 = X$. Если в заданном массиве C из m элементов нет отрицательных элементов, сформировать матрицу из k строк и m столбцов по правилу: первая строка – массив C , вторая корень квадратный из соответствующих элементов C , третья – корень кубический и так далее. Точность считается заданной.
28. Если в первой строке заданной матрицы все элементы меньше заданной величины ϵ , исключить ее из матрицы, иначе подсчитать число элементов, меньших ϵ , в каждой строке и определить строку с наибольшим количеством таких элементов.
29. Функция $F(x)$ определена с помощью ряда: $F(x) = \sum_{n=1}^{\infty} \frac{x^n}{n^3}$. Из заданного массива Y выделить и упорядочить по возрастанию только те элементы, для которых заданная точность вычисления $F(Y_i)$ достигается при суммировании не более m слагаемых.