**Пермский национальный исследовательский  
политехнический университет**

**О Т Ч Ё Т**

**по лабораторной работе №10**

Динамические массивы

Дисциплина: основы алгоритмизации и программирования

Вариант: 25

Выполнил работу

студент группы РИС-20-1б

Рябов Никита Андреевич

Проверила

Доцент кафедры ИТАС Полякова О.А.

Пермь, 2021

**Цель**

Организация динамических массивов.

**Постановка задачи**

Написать программу, в которой создаются динамические массивы и выполнить их обработку в соответствии со своим вариантом.

Сформировать массив строк. Добавить в него строку с заданным номером.

**Анализ задачи**

*Какие предстоит выполнить действия.*

Создать динамический массив заданной длины

int n=-1;

while (n<=0)

{

cout << "Введите длину массива\n";

cin >> n;

}

string \*mas=CreateArr(n);

Заполнить его с клавиатуры и распечатать

for (int i=0;i<n;i++) getline(cin,mas[i]);

PrintArr(mas,n);

Считать номер новой строки и саму строку

int num=-1;

string s;

while (num<=0 || num>n-1)

{

cout << "Введите номер\n";

cin >> num;

}

cout << "Введите строку\n";

cin.get();

getline(cin,s);

Вставить строку, распечатать массив и освободить память

mas=AddString(mas,num-1,s,&n);

PrintArr(mas,n);

DelArr(mas);

Функция CreateArr просто создает массив и возвращает указатель на него

string \*mas=new string[n];

return mas;

Функция PrintArr печатает массив в цикле for

void PrintArr(string mas[],int n)

{

cout << "Массив\n";

for (int i=0; i<n;i++) cout << mas[i] << endl;

}

Функция AddString создает второй массив, записывает в него старый массив и новую строку. Старый массив удаляется. Функция возвращяет указатель на новый массив.

string\* AddString(string mas[],int num, string s, int \*n)

{

string \*mas2=new string[\*n+1];

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

{

if (i<num) mas2[i]=mas[i];

else mas2[i+1]=mas[i];

}

mas2[num]=s;

delete [] mas;

\*n=\*n+1;

return mas2;

}

Функция DelArr освобождает память

void DelArr(string mas[])

{

delete [] mas;

}

*С каким типом данных действуем*

Динамический массив строк

*В каком виде представлены данные*

Строки, содержащие любые символы

*Какие поля*

Структуры в программе отсутствуют.

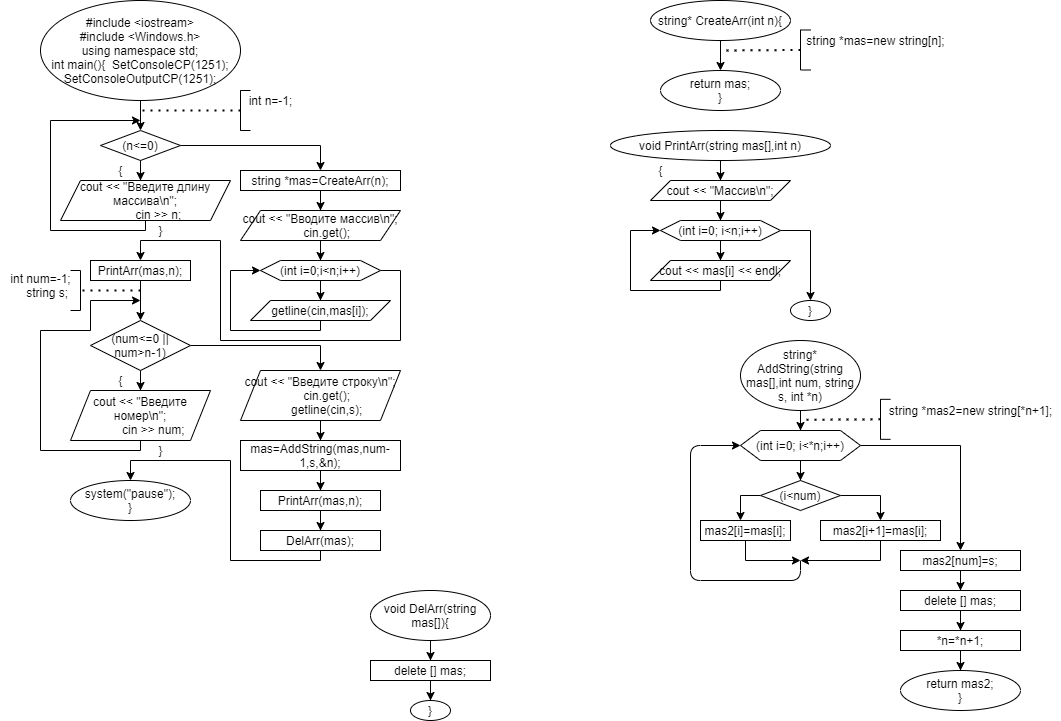
*Операторы ввода/вывода*

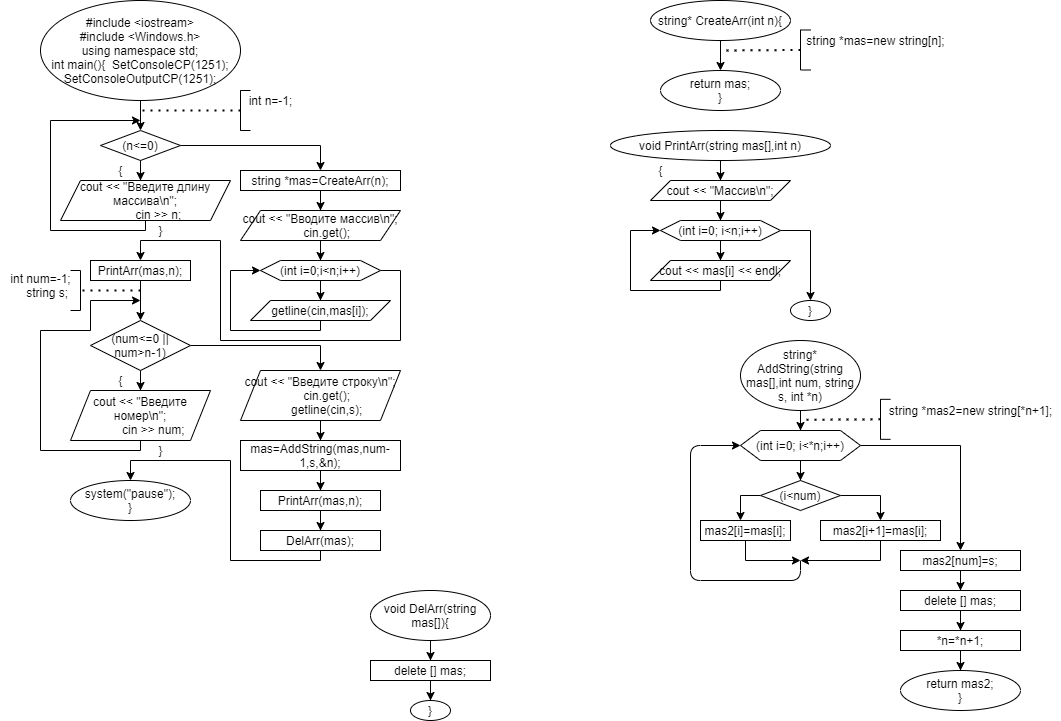
В основном используются операторы cin/cout. Исключение составляет ввод строк. Они могут содержать пробелы, и к ним применяется метод getline().

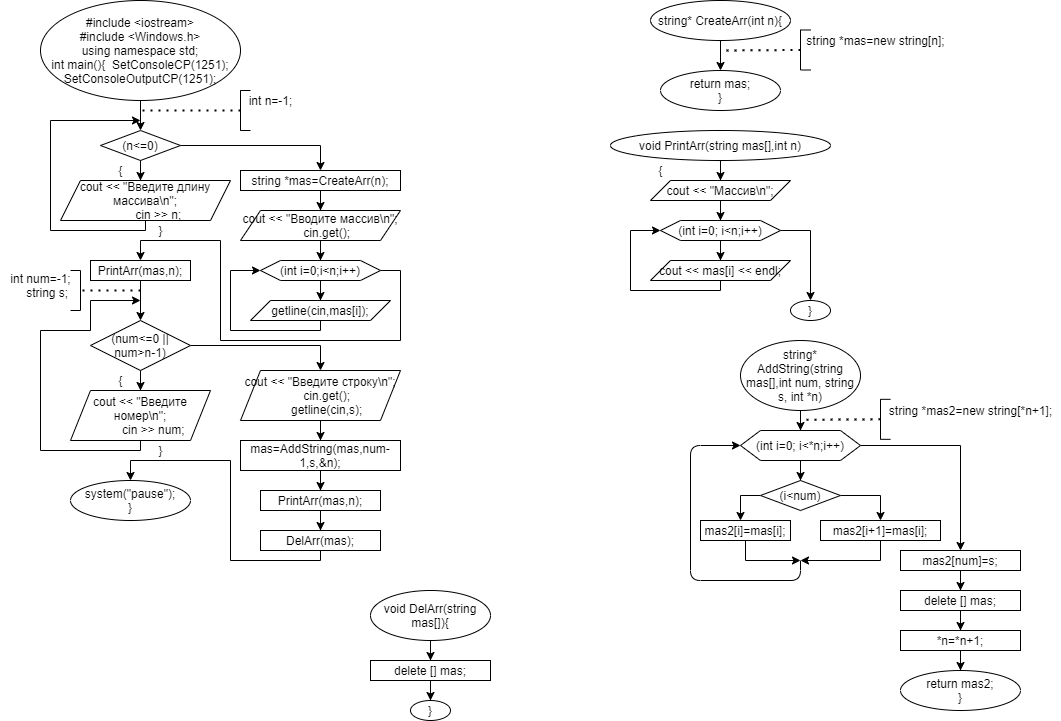
*Какие действия*

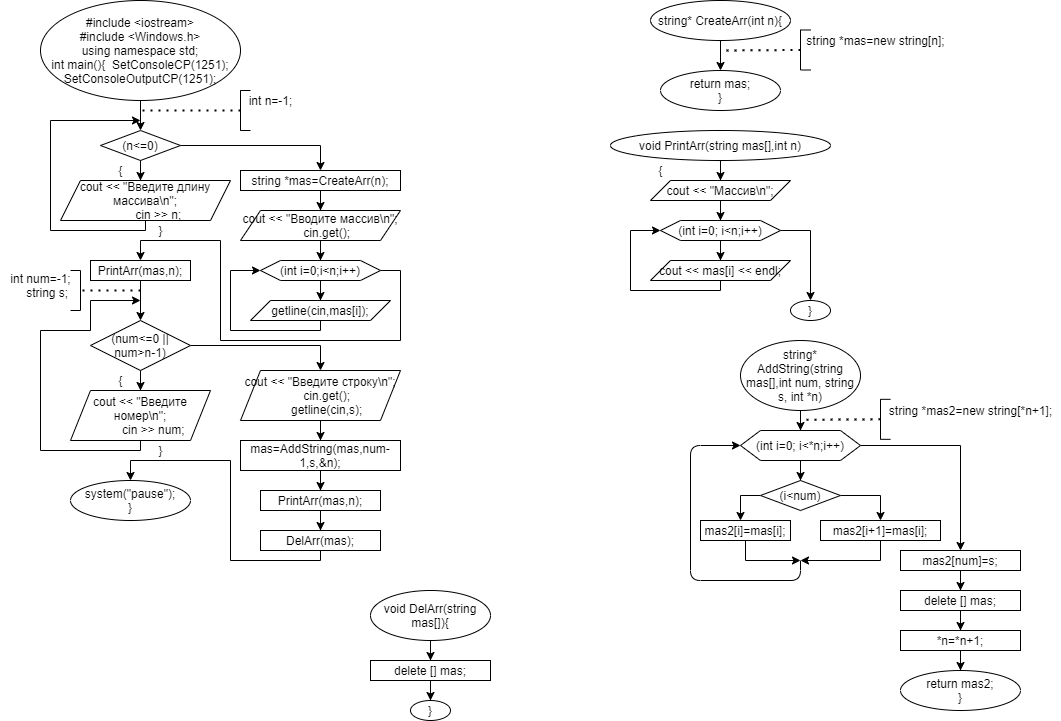
* Печать массива – for
* Создание нового массива – for/if
* Проверка корректности ввода – while
* Считывание массива – for

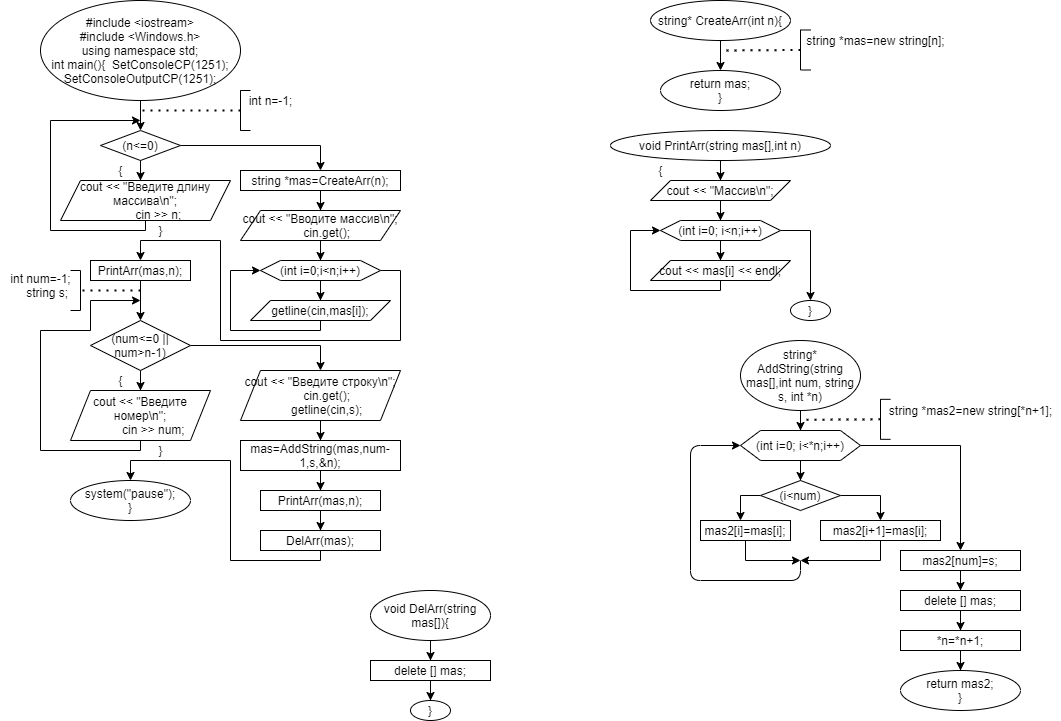
**Блок-схема**

****

****

****

****

****

**Код**

#include <iostream>

#include <Windows.h>

using namespace std;

string\* CreateArr(int n)

{

string \*mas=new string[n];

return mas;

}

void PrintArr(string mas[],int n)

{

cout << "Массив\n";

for (int i=0; i<n;i++) cout << mas[i] << endl;

}

string\* AddString(string mas[],int num, string s, int \*n)

{

string \*mas2=new string[\*n+1];

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

{

if (i<num) mas2[i]=mas[i];

else mas2[i+1]=mas[i];

}

mas2[num]=s;

delete [] mas;

\*n=\*n+1;

return mas2;

}

void DelArr(string mas[])

{

delete [] mas;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int n=-1;

while (n<=0)

{

cout << "Введите длину массива\n";

cin >> n;

}

string \*mas=CreateArr(n);

cout << "Вводите массив\n";

cin.get();

for (int i=0;i<n;i++) getline(cin,mas[i]);

PrintArr(mas,n);

int num=-1;

string s;

while (num<=0 || num>n-1)

{

cout << "Введите номер\n";

cin >> num;

}

cout << "Введите строку\n";

cin.get();

getline(cin,s);

mas=AddString(mas,num-1,s,&n);

PrintArr(mas,n);

DelArr(mas);

system("pause");

}

**Скриншоты**

****