



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

« МИРЭА Российский технологический университет »

РТУ МИРЭА

Институт Информационных технологий

Кафедра Вычислительной техники

УЧЕБНОЕ ЗАДАНИЕ

по дисциплине

« Объектно-ориентированное программирование »

Наименование задачи:

« Задача 4_3_1 »

С тудент группы

ИКБО-13-21

Дамарад Д.В.

Руководитель практики

Ассистент

Асадова Ю.С.

Работа представлена

«__»_____ 2022 г.

(подпись студента)

Оценка

(подпись руководителя)

Москва 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	
Постановка задачи.....	
Метод решения.....	
Описание алгоритма.....	
Блок-схема алгоритма.....	
Код программы.....	
Тестирование.....	
ЗАКЛЮЧЕНИЕ.....	
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ (ИСТОЧНИКОВ).....	

ВВЕДЕНИЕ

Постановка задачи

Создать базовый класс, содержащий только свойства. На его базе создать производный класс 1 и производный класс 2, содержащие только методы. В первом производном классе реализовать метод ввода размерности целочисленного массива и значений элементов массива. Во втором классе реализовать метод упорядочения целочисленного массива по не убыванию и метод вывода массива.

Посредством множественного наследования классов 1 и 2 создать класс 3. В классе 3 реализовать метод с алгоритмом:

1. Вызов метода ввода целочисленного массива.
2. Вызов метода вывода массива.
3. Вызов метода упорядочения целочисленного массива.
4. Вызов метода вывода массива.

Описание входных данных

Первая строка:

«размерность целочисленного массива»

Размерность массива натуральное число больше или равно 1 и меньше или равно 100.

Вторая строка:

«последовательность целых чисел»

Количество целых чисел равно размерности массива и разделены пробелами. Значение каждого числа принадлежит интервалу $[-100, 100]$.

Описание выходных данных

Первая строка:

Array dimension: «размерность целочисленного массива»

Вторая

строка:

The original array: «последовательность целых чисел»

Для вывода каждого числа выделяется поле из 5 позиции.

Третья

строка:

An ordered array: «последовательность целых чисел»

Для вывода каждого числа выделяется поле из 5 позиции.

Метод решения

Для решения поставленной задачи используются:

- Библиотека стандартных объектов потока ввода и вывода `cin` и `cout` соответственно, для ввода и вывода на экран.
- Объект `obj` класса `Class3`.

Класс `Base`:

- Поля:
 - Поле размера массива:
 - Наименование - `size`.
 - Тип - целочисленный.
 - Модификатор доступа - `protected`.
 - Поле динамического массива:
 - Наименование - `arr`.
 - Тип - целочисленный указатель.
 - Модификатор доступа - `protected`.

Класс `Class1`, наследуемый от виртуального класса `Base`:

- Методы:
 - Метод `InputSizeNums`.
 - Функционал - ввод размера массива и его заполнение.

Класс `Class2`, наследуемый от виртуального класса `Base`:

- Методы:
 - Метод `Sort`.
 - Функционал - упорядочивание целочисленного массива по неубыванию.

- Метод Print.
 - Функционал - вывод значений целочисленного массива.

Класс Class3, наследуемый от виртуальных классов Class1 и Class2:

- Методы:
 - Метод Method:
 - Функционал - создание массива и его заполнение, вывод массива до упорядочивания и после.

Описание алгоритма

Согласно этапам разработки, после определения необходимого инструментария в разделе «Метод», составляются подробные описания алгоритмов для методов классов и функций.

Функция: main

Функционал: Основной алгоритм программы

Параметры: Отсутствуют

Возвращаемое значение: Целочисленное значение - код возврата

Алгоритм функции представлен в таблице 1.

Таблица 1. Алгоритм функции main

№	Предикат	Действия	№ перехода	Комментарий
1		Создание объекта obj класса Class3	2	
2		Вызов метода Method объекта obj	Ø	

Класс объекта: Class1

Модификатор доступа: protected

Метод: InputSizeNum

Функционал: Ввод размера массива и его заполнение

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 2.

Таблица 2. Алгоритм метода InputSizeNum класса Class1

№	Предикат	Действия	№ перехода	Комментарий
1		Считывание с клавиатуры значения переменной size	2	
2		Вывод: "Array dimension: " (size) и перевод строки	3	
3		Инициализация динамического массива arr размером size	4	
4		Объявление и инициализация переменной i=0	5	
5	i меньше size	Считывание с клавиатуры значения arr[i]	6	
			Ø	
6		Увеличение i на 1	5	

Класс объекта: Class2

Модификатор доступа: protected

Метод: Sort

Функционал: Упорядочивание целочисленного массива по неубыванию

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 3.

Таблица 3. Алгоритм метода Sort класса Class2

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление переменных i и j и инициализация i=0	2	
2	i меньше size	Инициализация j=0	3	

			Ø	
3	j меньше size		4	
		Увеличение i на 1	2	
4	arr[i] меньше, чем arr[j]	Меняем местами arr[i] и arr[j]	5	
			5	
5		Увеличение j на 1	3	

Класс объекта: Class2

Модификатор доступа: protected

Метод: Print

Функционал: Вывод значения целочисленного массива

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 4.

Таблица 4. Алгоритм метода Print класса Class2

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление и инициализация переменной i=0	2	
2	i меньше size	Вывод arr[i]	2	
			Ø	

Класс объекта: Class3

Модификатор доступа: public

Метод: Method

Функционал: Создание массива и его заполнение, вывод массива до

упорядочивания и после

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 5.

Таблица 5. Алгоритм метода Method класса Class3

№	Предикат	Действия	№ перехода	Комментарий
1		Вызов метода InputSizeNum	2	
2		Вывод: "The original array:"	3	
3		Вызов метода Print	4	
4		Перевод строки	5	
5		Вызов метода Sort	6	
6		Перевод строки	7	
7		Вывод: "An ordered array:"	8	
8		Вызов метода Print	Ø	

Блок-схема алгоритма

Представим описание алгоритмов в графическом виде на рисунках ниже.

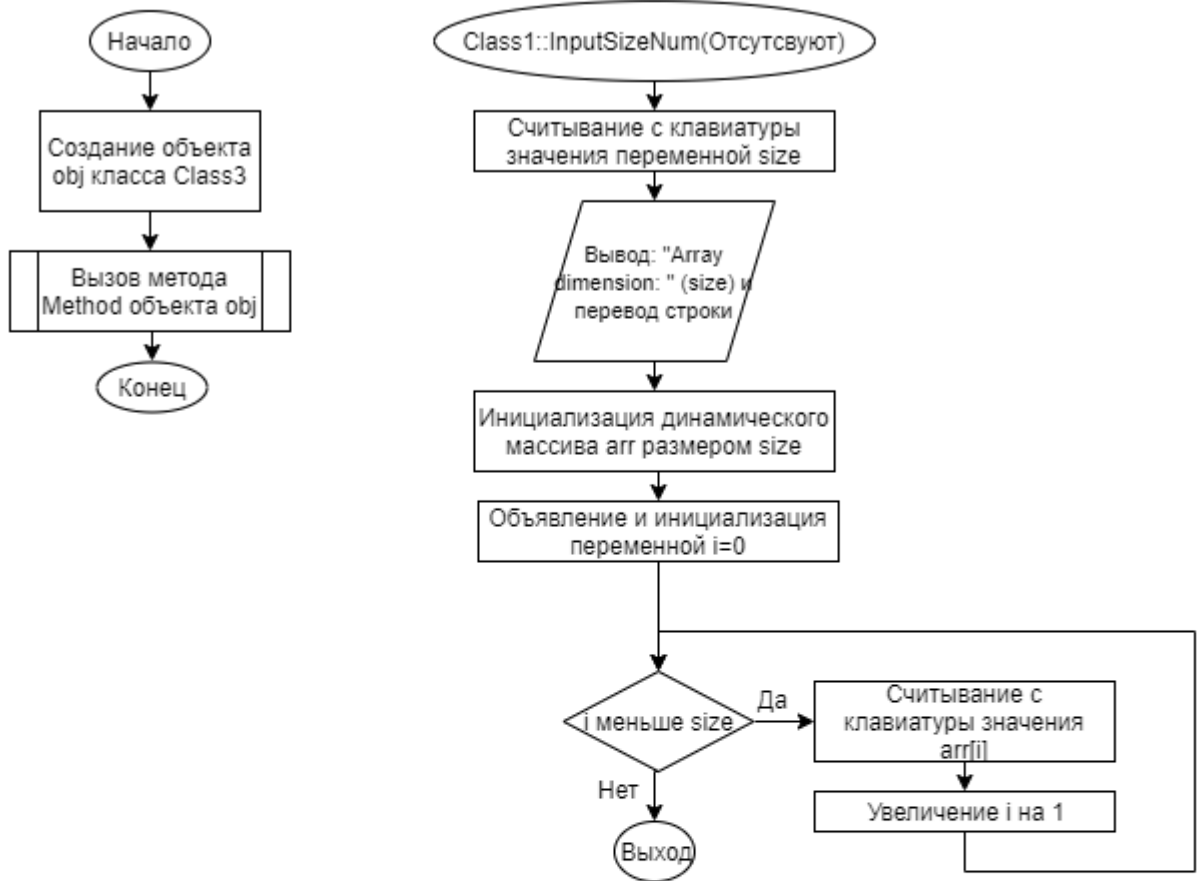


Рис. 1. Блок-схема алгоритма.

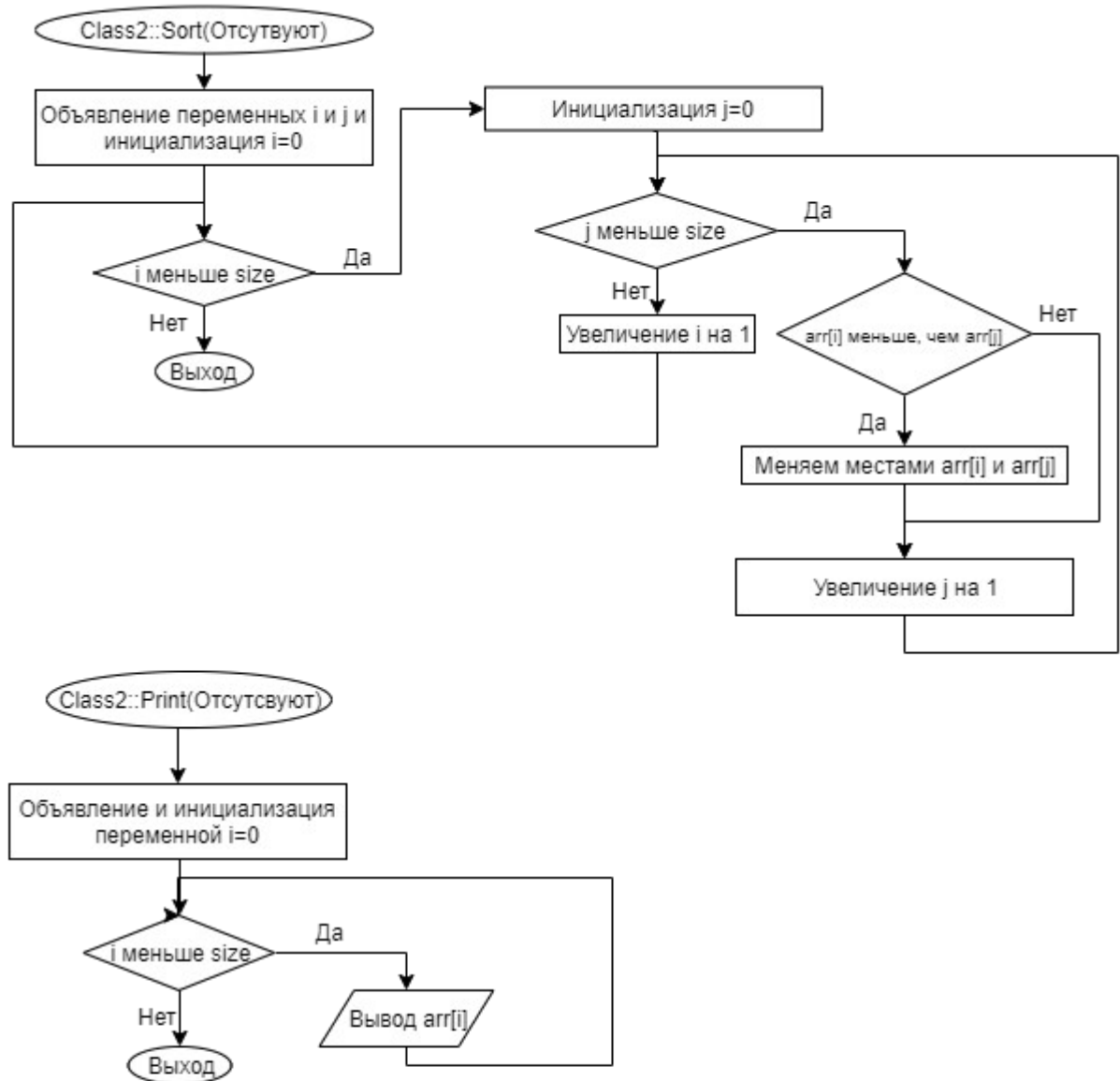


Рис. 2. Блок-схема алгоритма.

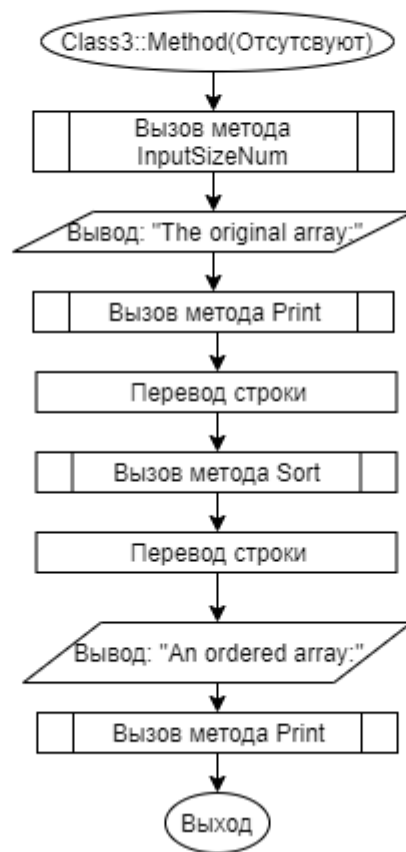


Рис. 3. Блок-схема алгоритма.

Код программы

Программная реализация алгоритмов для решения задачи представлена ниже.

Файл Base.h

```
#ifndef BASE_  
#define BASE_  
#include <iomanip>  
#include <iostream>  
using namespace std;  
class Base{  
protected:  
    int size;  
    int* arr;  
};  
#endif
```

Файл Class1.cpp

```
#include "Base.h"  
#include "Class1.h"  
void Class1::InputSizeNum(){  
    cin>>size;  
    cout<<"Array dimension: "<<size<<endl;  
    arr=new int[size];  
    for (int i=0;i<size;i++){  
        cin>>arr[i];  
    }  
}
```

Файл Class1.h

```
#ifndef CLASS1_  
#define CLASS1_  
#include "Base.h"  
class Class1:public virtual Base{  
protected:  
    void InputSizeNum();  
};  
#endif
```

Файл Class2.cpp

```
#include "Base.h"
#include "Class2.h"
void Class2::Sort(){
    for (int i=0;i<size;i++){
        for (int j=0;j<size;j++){
            if (arr[i]<arr[j]){
                swap(arr[i],arr[j]);
            }
        }
    }
}
void Class2::Print(){
    for (int i=0;i<size;i++){
        cout<<setw(5)<<arr[i];
    }
}
```

Файл Class2.h

```
#ifndef CLASS2_
#define CLASS2_
#include "Base.h"
class Class2:public virtual Base{
protected:
    void Sort();
    void Print();
};
#endif
```

Файл Class3.cpp

```
#include "Class1.h"
#include "Class2.h"
#include "Class3.h"
void Class3::Method(){
    Class1::InputSizeNum();
    cout<<"The original array:";
    Class2::Print();
    cout<<endl;
    Class2::Sort();
    cout<<"An ordered array:";
    Class2::Print();
}
```


Файл Class3.h

```
#ifndef CLASS3_  
#define CLASS3_  
#include "Class1.h"  
#include "Class2.h"  
class Class3:public Class1, public Class2{  
public:  
    void Method();  
};  
#endif
```

Файл main.cpp

```
#include "Class3.h"  
int main()  
{  
    Class3 obj;  
    obj.Method();  
    return(0);  
}
```

Тестирование

Результат тестирования программы представлен в следующей таблице.

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
5 5 4 3 2 1	Array dimension: 5 The original array: 5 4 3 2 1 An ordered array: 1 2 3 4 5	Array dimension: 5 The original array: 5 4 3 2 1 An ordered array: 1 2 3 4 5
3 7 9 3	Array dimension: 3 The original array: 7 9 3 An ordered array: 3 7 9	Array dimension: 3 The original array: 7 9 3 An ordered array: 3 7 9

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ (ИСТОЧНИКОВ)

1. Васильев А.Н. Объектно-ориентированное программирование на C++. Издательство: Наука и Техника. Санкт-Петербург, 2016г. 543 стр.
2. Шилдт Г. C++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2017. — 624 с.
3. Методическое пособие для проведения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] – URL: https://mirea.aco-avrrora.ru/student/files/methodichescoe_posobie_dlya_laboratornyh_rabot_3.pdf (дата обращения 05.05.2021).
4. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: https://mirea.aco-avrrora.ru/student/files/Prilozheniye_k_methodichke.pdf (дата обращения 05.05.2021).
5. Видео лекции по курсу «Объектно-ориентированное программирование» [Электронный ресурс]. АСО «Аврора».
6. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие /Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).