



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\_4\_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 метод наименованием `in_out` без параметров.

Метод организует ввод исходных данных: размерности целочисленного массива и значений элементов целочисленного массива.

На базе базового класса создать производный класс 1 содержащий метод целого типа наименованием `func`.

Метод от значения первого элемента целочисленного массива последовательно вычитает значения остальных элементов. Полученный результат возвращает.

На базе класса 1 создать производный класс 2 содержащий метод целого типа наименованием `func`.

Метод суммирует элементы целочисленного массива. Полученный результат возвращает.

На базе класса 2 создать производный класс 3 содержащий метод наименованием `in_out` без параметров для вывода исходных данных и вычисленных значений.

В основной программе использовать один указатель на объект класса и один объект.

В основной программе реализовать алгоритм:

1. Объявить указатель на объект класса.
2. Создать объект оператором `new`.
3. Ввод исходных данных: размерности целочисленного массива и значений элементов данного массива.
4. Вычисление значение вычитания от первого элемента массива всех

остальных.

5. Вычисление суммы элементов массива.

6. Вывод полученных результатов.

### **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:**

Min: «значение вычитания из первого элемента массива всех остальных»

**Четвертая строка:**

Sum: «значение суммы элементов массив»

## Метод решения

Для решения поставленной задачи используются:

- Библиотека стандартных потоков ввода и вывода `cin` и `cout` соответственно для ввода и вывода на экран.
- Указатель `obj` на объект класса `Class3`.

Класс `Base`:

- Поля:
  - Поле размера массива:
    - Наименование - `size`;
    - Тип - `int`;
    - Модификатор доступа - `protected`.
  - Поле целочисленного указателя на массив:
    - Наименование - `arr`;
    - Тип - `int`;
    - Модификатор доступа - `protected`.
- Методы:
  - Метода `in_out`:
    - Функционал - ввод размерности массива и его элементов.

Класс `Class1`:

- Методы:
  - Метод `func`:
    - Функционал - вычитает из первого элемента массива последующие элементы.

Класс Class2:

- Методы:
  - Метод func:
    - Функционал - суммирует элементы массива.

Класс Class3:

- Методы:
  - Метод in\_out:
    - Функционал - вывод исходных данных и вычисленных значений.

№	Имя класса	Классы - наследники	Модификатор доступа при наследовании	Описание	Номер	Комментарий
1	Base	Class1	public	Базовый класс в иерархии классов	2	
2	Class1	Class2	public	Класс для вычисления разности первого элемента и последующих	3	
3	Class2	Class3	public	Класс для вычисления суммы всех элементов	4	
4	Class3			Класс для вывода исходных данных и вычислений значений		



## 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 2.

Таблица 2. Алгоритм функции main

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление указателя obj на класс Class3	2	
2		Инициализация указателя obj	3	
3		Вызов метода in_out класса Base объекта obj	4	
4		Вызов метода in_out класса Class3 объекта obj	5	
5		Удаление объекта obj	Ø	

Класс объекта: Base

Модификатор доступа: public

Метод: in\_out

Функционал: Ввод размерности массива и его элементов

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 3.

Таблица 3. Алгоритм метода in\_out класса Base

№	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 размера 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: Class1

Модификатор доступа: public

Метод: func

Функционал: Вычисление разности между первым и последующими элементами

Параметры: Отсутствуют

Возвращаемое значение: Целочисленное значение - разность первого и последующих элементов

Алгоритм метода представлен в таблице 4.

Таблица 4. Алгоритм метода func класса Class1

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление переменной res и инициализация res=arr[0]	2	
2		Объявление и инициализация переменной i=1	3	
3	i меньше size	Вычитание из res элемента arr[i]	4	
		Возврат res	Ø	
4		Увеличение i на 1	3	

Класс объекта: Class2

Модификатор доступа: public

Метод: func

Функционал: Суммирование всех элементов массива

Параметры: Отсутствуют

Возвращаемое значение: Целочисленное значение - сумма всех элементов

Алгоритм метода представлен в таблице 5.

Таблица 5. Алгоритм метода func класса Class2

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление и инициализация переменной res=0	2	
2		Объявление и инициализация переменной i=0	3	
3	i меньше size	Суммирование res и arr[i]	4	
		Возврат res	Ø	

4		Увеличение i на 1	Ø	
---	--	-------------------	---	--

Класс объекта: Class3

Модификатор доступа: public

Метод: in\_out

Функционал: Вывод элементов массива на экран и значений вычислений

Параметры: Отсутствуют

Возвращаемое значение: void

Алгоритм метода представлен в таблице 6.

Таблица 6. Алгоритм метода in\_out класса Class3

№	Предикат	Действия	№ перехода	Комментарий
1		Вывод: "The original array:"	2	
2		Объявление и инициализация переменной i=0	3	
3	i меньше size	Вывод arr[i]	4	
		Вывод: "Min:" (значение функции func класса Class1) и перевод строки	5	
4		Увеличение i на 1	3	
5		Вывод: "Sum:" (значение функции func класса Class2)	Ø	

## Блок-схема алгоритма

Представим описание алгоритмов в графическом виде на рисунках ниже.

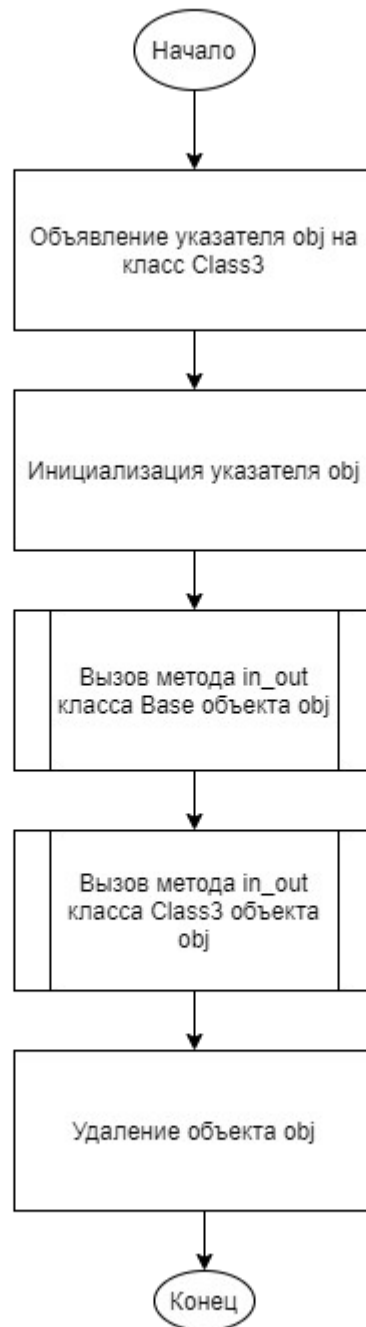


Рис. 1. Блок-схема алгоритма.

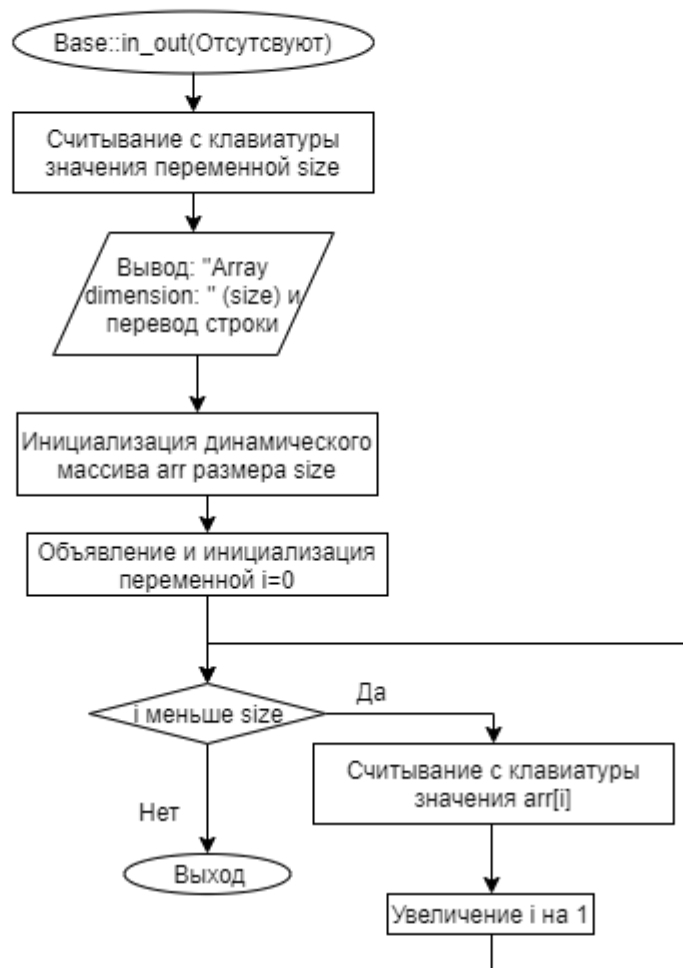


Рис. 2. Блок-схема алгоритма.

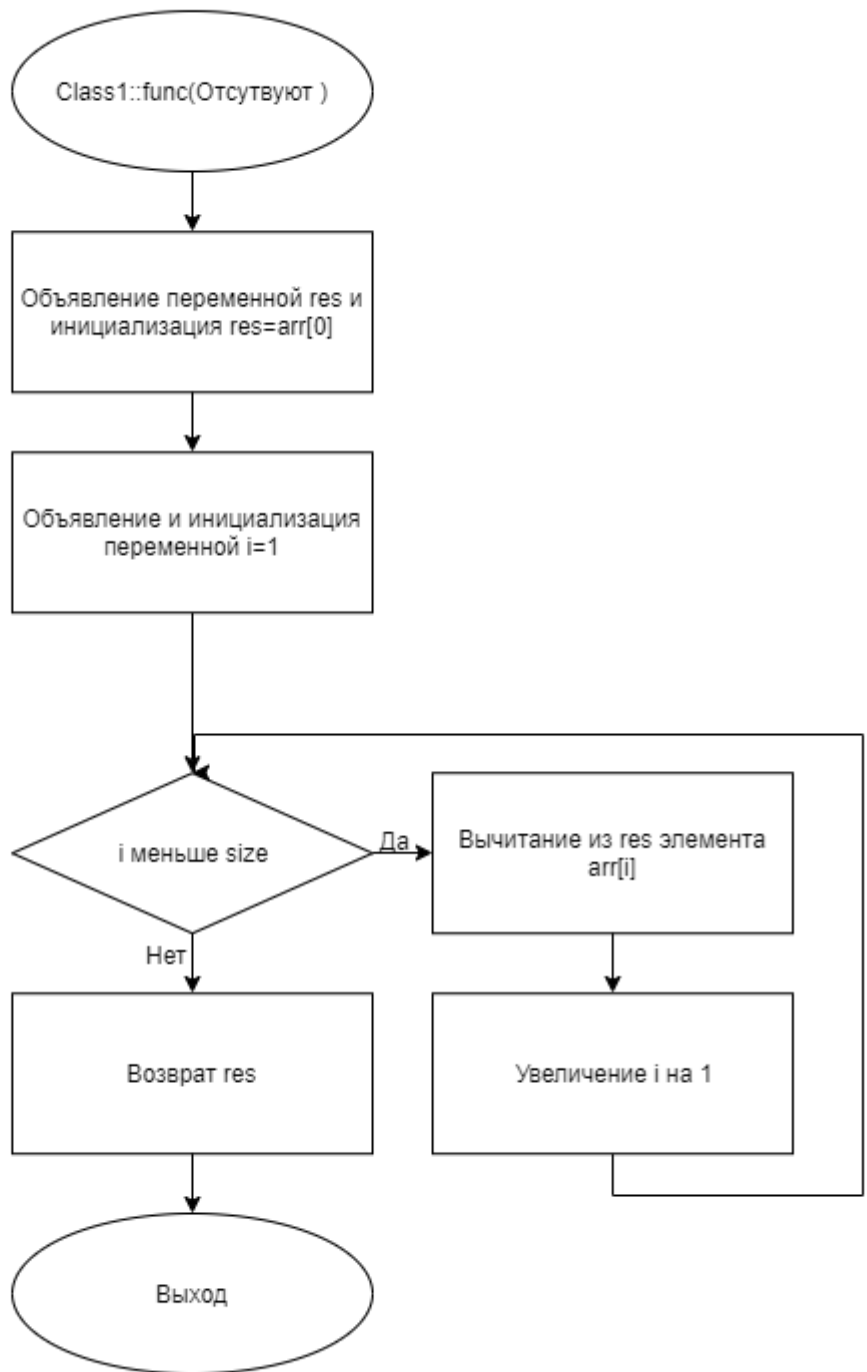


Рис. 3. Блок-схема алгоритма.

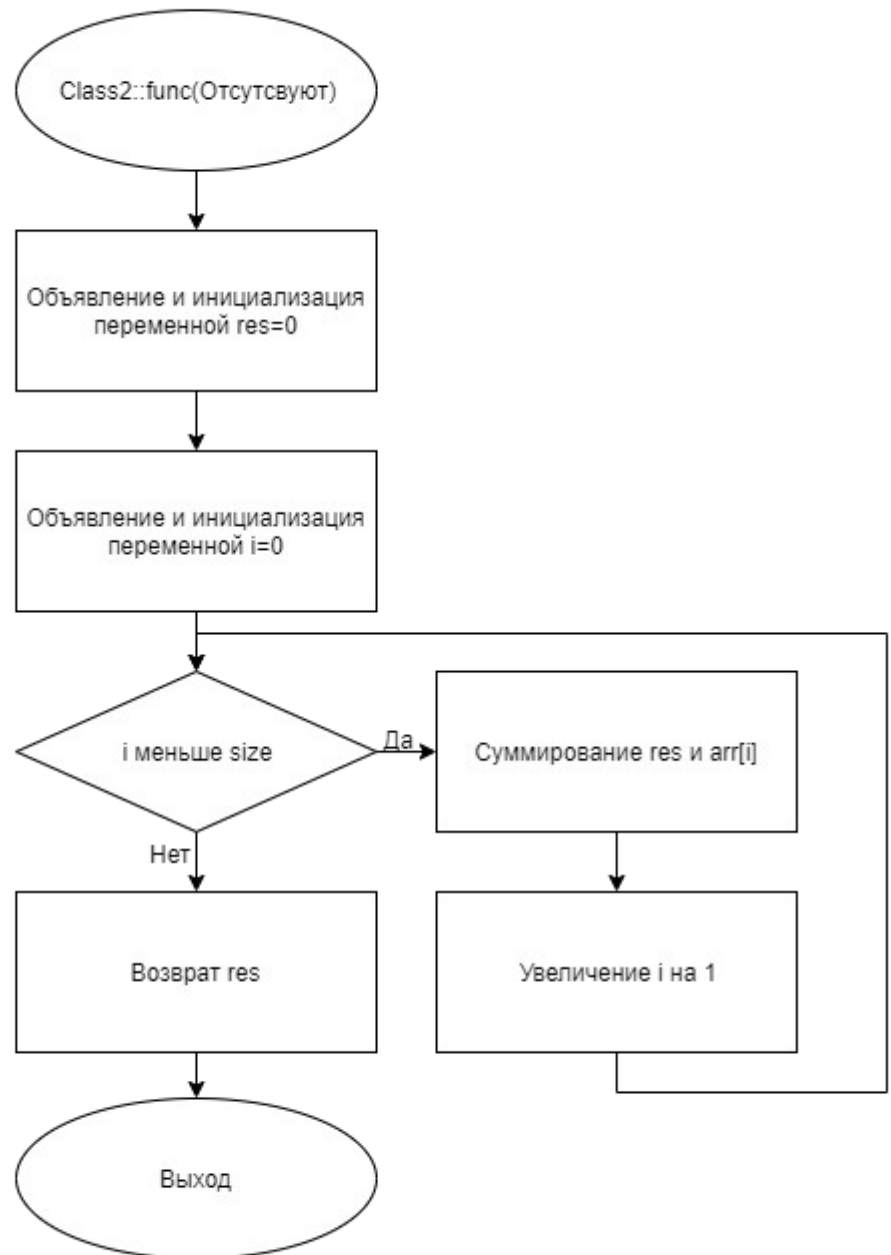


Рис. 4. Блок-схема алгоритма.



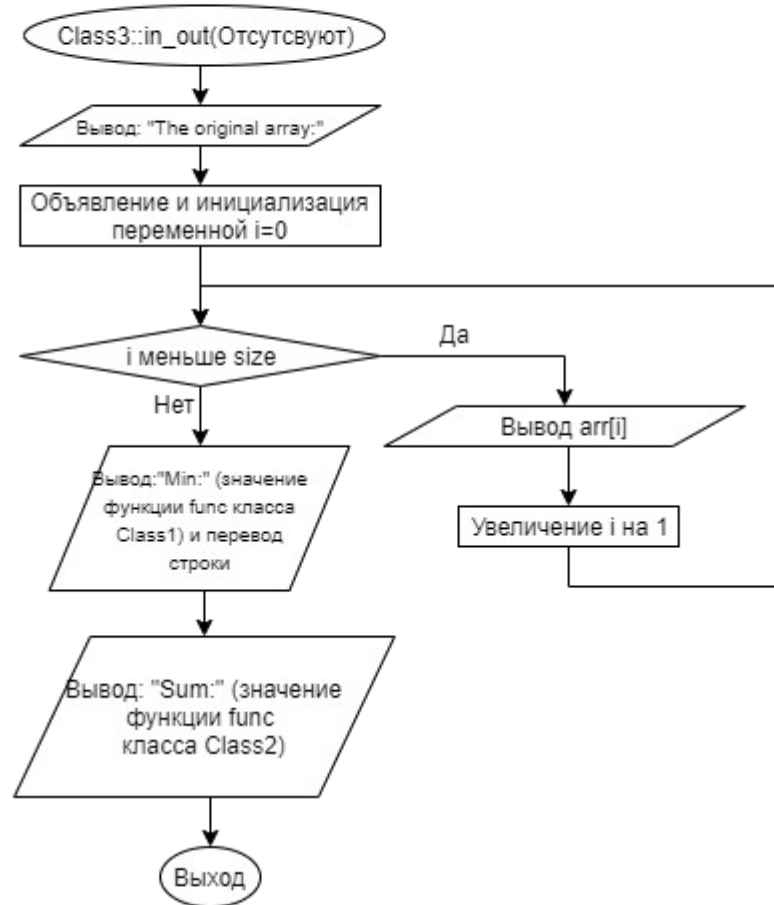


Рис. 5. Блок-схема алгоритма.

## Код программы

Программная реализация алгоритмов для решения задачи представлена ниже.

### Файл Base.cpp

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

### Файл Base.h

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

### Файл Class1.cpp

```
#include "Base.h"
#include "Class1.h"
int Class1::func(){
    int res=arr[0];
    for (int i=1;i<size;i++){
        res-=arr[i];
    }
    return res;
}
```

## Файл Class1.h

```
#ifndef CLASS1_  
#define CLASS1_  
#include "Base.h"  
class Class1:public Base{  
public:  
    int func();  
};  
#endif
```

## Файл Class2.cpp

```
#include "Class1.h"  
#include "Class2.h"  
int Class2::func(){  
    int res=0;  
    for (int i=0;i<size;i++){  
        res+=arr[i];  
    }  
    return res;  
}
```

## Файл Class2.h

```
#ifndef CLASS2_  
#define CLASS2_  
#include "Class1.h"  
class Class2:public Class1{  
public:  
    int func();  
};  
#endif
```

## Файл Class3.cpp

```
#include "Class2.h"  
#include "Class3.h"  
void Class3::in_out(){  
    cout<<"The original array:";
```

```

        for (int i=0;i<size;i++){
            cout<<setw(5)<<arr[i];
        }
        cout<<"\nMin: "<<::Class1::func()<<endl;
        cout<<"Sum: "<<::Class2::func();
    }
}

```

### **Файл Class3.h**

```

#ifndef CLASS3_
#define CLASS3_
#include "Class2.h"
using namespace std;
class Class3:public Class2{
public:
    void in_out();
};
#endif

```

### **Файл main.cpp**

```

#include "Class3.h"
int main(){
    Class3* obj;
    obj=new Class3;
    obj->Base::in_out();
    obj->in_out();
    delete obj;
    return 0;
}

```

## Тестирование

Результат тестирования программы представлен в следующей таблице.

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
3 9 8 7	Array dimension: 3 The original array: 9 8 7 Min: -6 Sum: 24	Array dimension: 3 The original array: 9 8 7 Min: -6 Sum: 24
5 1 2 3 4 5	Array dimension: 5 The original array: 1 2 3 4 5 Min: -13 Sum: 15	Array dimension: 5 The original array: 1 2 3 4 5 Min: -13 Sum: 15

## **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](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](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).