



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 3_1_2 »

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 объект первого типа, у которого одно целочисленное свойство. Значение данного свойства определяется посредством параметризованного конструктора.

Создать объект второго типа, у которого две целочисленных свойства. Значение данных свойств определяется посредством метода объекта. Реализовать дружественную функцию, которая находит максимальное значение полей объекта первого типа и полей объекта второго типа.

Написать программу:

1. Вводит значение для поля объекта первого типа.
2. Создает объект первого типа.
3. Вводит значения полей для полей объекта второго типа.
4. Создает объект второго типа.
5. Определяет значения полей объекта второго типа.
6. Определяет максимальное значение полей, созданных двух объектов разного типа посредством дружественной функции.
7. Выводит полученный результат.

Описание входных данных

Первая строка:

«целое число в десятичном формате»

Вторая строка:

«целое число в десятичном формате»_«целое число в десятичном формате»

Описание выходных данных

Первая строка, с первой позиции:

max = «целочисленное значение в десятичном формате»

Метод решения

Для решения поставленной задачи используется:

- Библиотека стандартных объектов потока ввода/вывода (cin/cout).
- Используется для считывания с клавиатуры и для вывода на экран.
- Условный оператор для сравнения чисел. Используется для сравнения полей объектов.
- Объекты obj1 и obj2 классов Firsttype и Secondtype соответственно.
- Собственно написанная функция Compare для сравнения полей объектов.

Класс Firsttype:

- Поле:
 - Наименование - a;
 - Тип - целочисленный;
 - Модификатор доступа - private.
- Методы:
 - Метод Firsttype:
 - Функционал - параметризованный конструктор
- Дружественные функции:
 - Функция Compare

Класс Secondtype:

- Поля:
 - Наименование - a,b;
 - Тип - целочисленный;

- Модификатор доступа - private.
- Методы:
 - Метод Set_a_b:
 - Функционал - сеттер для полей a,b объекта
- Дружественные функции:
 - Функция Compare

Функция Compare:

- Параметры:
 - Наименование - x,y;
 - Тип - пользовательский типы данных Firsttype, Secondtype;
- Семантика функций:
 - Алгоритм для сравнения полей объектов

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		Объявление целочисленных переменных value1,value2	2	
2		Считывание с клавиатуры значения value1	3	
3		Создание объекта obj1 типа Firsttype с передачей в конструктор значения value1	4	
4		Считывание с клавиатуры значения value1, value2	5	
5		Создание объекта obj2 типа Secondtype	6	
6		Вызов метода Set_a_b объекта obj2 с переменными value1, value2	7	
7		Вызов функции Compare с передачей объектов obj1, obj2	Ø	

Функция: Compare

Функционал: Сравнение полей объектов

Параметры: Ссылка на объект х типа Firsttype, ссылка на объект н типа Secondtype

Возвращаемое значение: void

Алгоритм функции представлен в таблице 2.

Таблица 2. Алгоритм функции Compare

№	Предикат	Действия	№ перехода	Комментарий
1		Объявление целочисленной переменной max	2	
2	Поле а объекта у больше поля объекта у	Присвоение переменной max значения поля а объекта у	3	
		Присвоение переменной max значения поля b объекта у	3	
3	Поле а объекта х больше max	Присвоение переменной max значения поля а объекта х	4	
			4	
4		Вывод на экран "max = "и значения max	∅	

Класс объекта: Firsttype

Модификатор доступа: public

Метод: Firsttype

Функционал: Параметризированный конструктор

Параметры: Целочисленная переменная a

Возвращаемое значение: void

Алгоритм метода представлен в таблице 3.

Таблица 3. Алгоритм метода Firsttype класса Firsttype

№	Предикат	Действия	№ перехода	Комментарий
1		Присвоение полю a значения переменной a	Ø	

Класс объекта: Secondtype

Модификатор доступа: public

Метод: Set_a_b

Функционал: Сеттер объекта

Параметры: Целочисленные переменные a,b

Возвращаемое значение: void

Алгоритм метода представлен в таблице 4.

Таблица 4. Алгоритм метода Set_a_b класса Secondtype

№	Предикат	Действия	№ перехода	Комментарий
1		Присвоение полю a значения переменной a	2	
2		Присвоение полю b значения переменной b	Ø	

Блок-схема алгоритма

Представим описание алгоритмов в графическом виде на рисунках ниже.

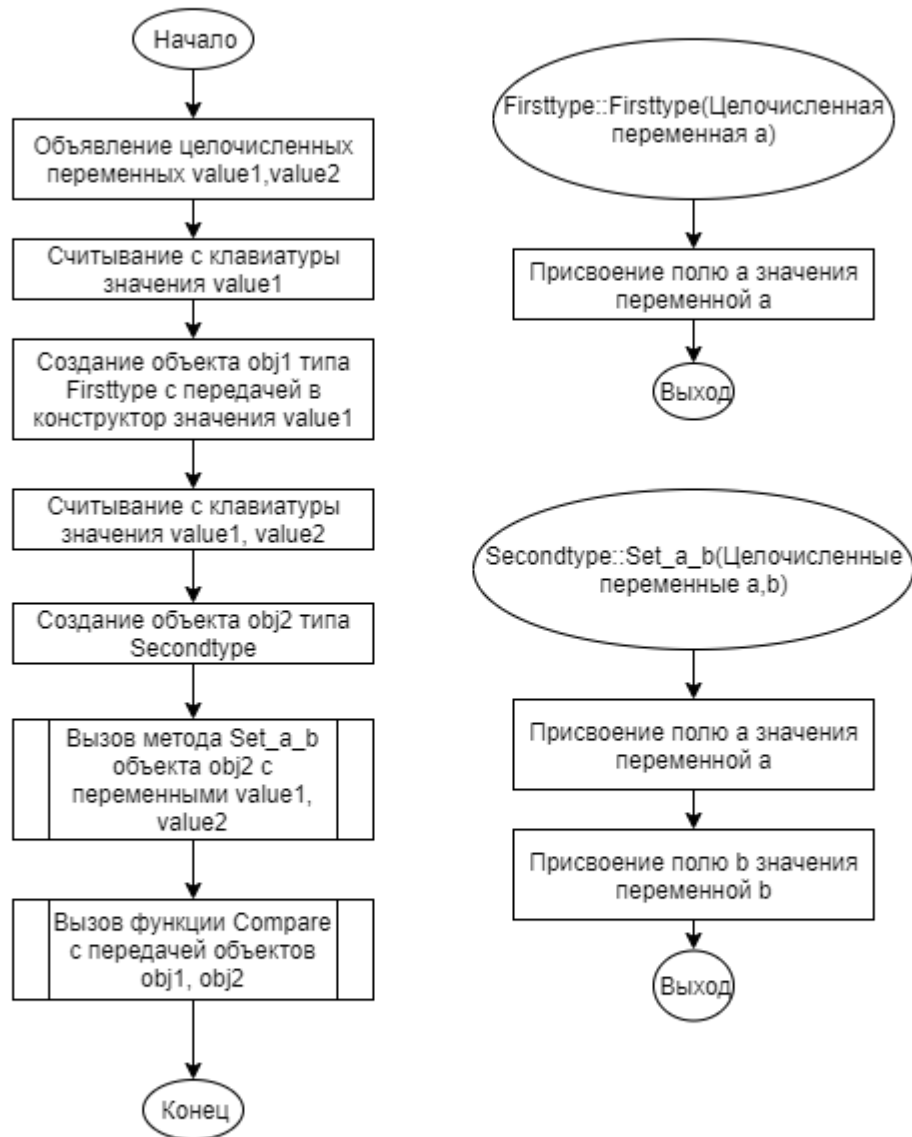


Рис. 1. Блок-схема алгоритма.

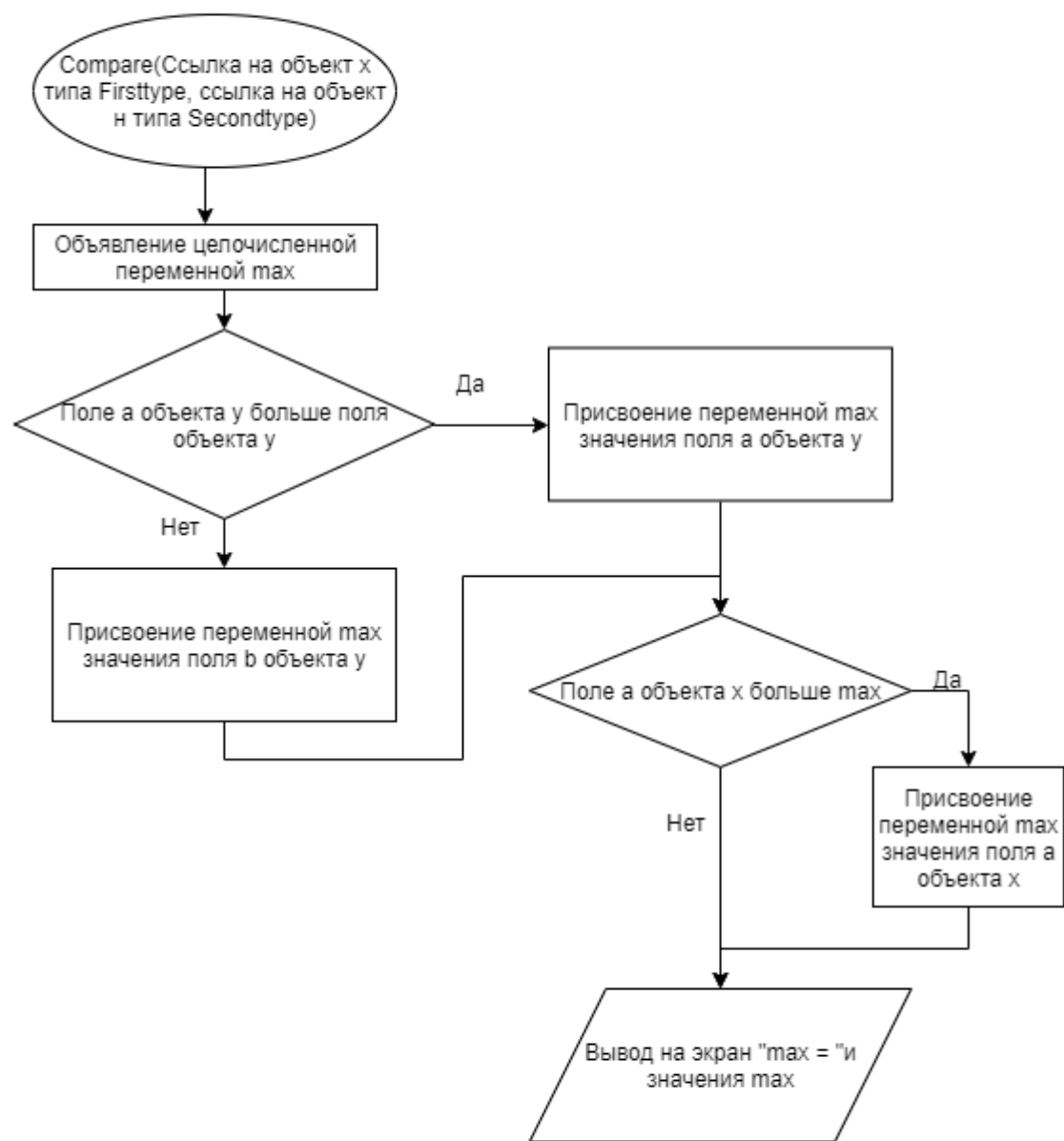


Рис. 2. Блок-схема алгоритма.

Код программы

Программная реализация алгоритмов для решения задачи представлена ниже.

Файл Firsttype.cpp

```
#include "Firsttype.h"
#include <iostream>
Firsttype::Firsttype(int a) {
    this->a = a;
}
```

Файл Firsttype.h

```
#ifndef _FIRSTTYPE_H
#define _FIRSTTYPE_H
class Secondtype; //прототип класса Secondtype
class Firsttype
{
private:
    int a;
public:
    Firsttype(int a); //конструктор объекта типа firsttype
    friend void Compare(Firsttype& x, Secondtype& y); //Дружественная
    функция для сравнения полей объектов разных типов
};
#endif
```

Файл main.cpp

```
#include <iostream>
#include "Firsttype.h"
#include "Secondtype.h"
using namespace std;
void Compare(Firsttype& x, Secondtype& y) {
    int max;
    if (y.a > y.b) max = y.a; //если поле a больше поля b объекта y то
    записываем в max значение поля a
    else max = y.b; //иначе записываем в max значение поля b
    if (x.a > max) max = x.a; //если поле a объекта x больше max то
    записываем в max значение поля a
    cout << "max = " << max;
}
int main() {
    int value1, value2;
```

```

        cin >> value1;
        Firsttype obj1(value1);
        cin >> value1 >> value2;
        Secondtype obj2;
        obj2.Set_a_b(value1, value2);
        Compare(obj1, obj2);
    }

```

Файл Secondtype.cpp

```

#include "Secondtype.h"
#include <iostream>
void Secondtype::Set_a_b(int a,int b) {
    this->a = a;
    this->b = b;
}

```

Файл Secondtype.h

```

#ifndef _SECONDTYPE_H
#define _SECONDTYPE_H
class Firsttype; //прототип класса Firsttype
class Secondtype
{
private:
    int a;
    int b;
public:
    void Set_a_b(int a, int b); //сеттер для объекта типа secondtype
    friend void Compare(Firsttype& x, Secondtype& y);
    //Дружественная функция для сравнения полей объектов
};
#endif

```

Тестирование

Результат тестирования программы представлен в следующей таблице.

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
-3 -2 -1	max = -1	max = -1
4 5 6	max = 6	max = 6

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).