



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_2_15_1 »

С тудент группы

ИКБО-26-20

Мажарова А.А.

Руководитель практики

Ассистент

Тимаков К.А.

Работа представлена

« ___ » _____ 2021 г.

(подпись студента)

Оценка

(подпись руководителя)

Москва 2021

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

Создадим 2 класса

1.cl1

Свойства:

целочисленная переменная

Функционал:

конструктор-просто получает значения

дружественная функция:

через условные операторы 'if' проверим какое число больше всех и при помощи 'cout' выведем наибольшее число на экран

2.cl 2

Свойства:

2 целочисленных переменных

Функционал:

метод input-приравнивает значения в закрытые поля данных

Дружественная функция-сравнивает 3 переменные при помощи условного оператора 'if' и при помощи объекта стандартного потока вывода 'cout' выводит наибольшее значение

Основная функция программы:

ввод значение, создание объектов класс и вызов дружественной функции

Описание алгоритма

Класс объекта: cl1

Модификатор доступа: public

Метод: c1

Функционал: конструктор класса

Параметры: a

Возвращаемое значение: нет

| № | Предикат | Действия | № перехода | Комментарий |
|---|----------|----------|------------|-------------|
| 1 | | A=a | ∅ | |

Класс объекта: c12

Модификатор доступа: public

Метод: input

Функционал: присвоение данных

Параметры: b,c

Возвращаемое значение: нет

| № | Предикат | Действия | № перехода | Комментарий |
|---|----------|--|------------|-------------|
| 1 | | присвоить введенные данные закрытому типу класса | ∅ | |

Функция: max

Функционал: вычисляет наибольшее число

Параметры: ob1,ob2

Возвращаемое значение: 0

| № | Предикат | Действия | № перехода | Комментарий |
|---|----------|---|------------|-------------|
| 1 | | проверить наибольшее и вывести его на экран | ∅ | |

Функция: main

Функционал: основная функция программы

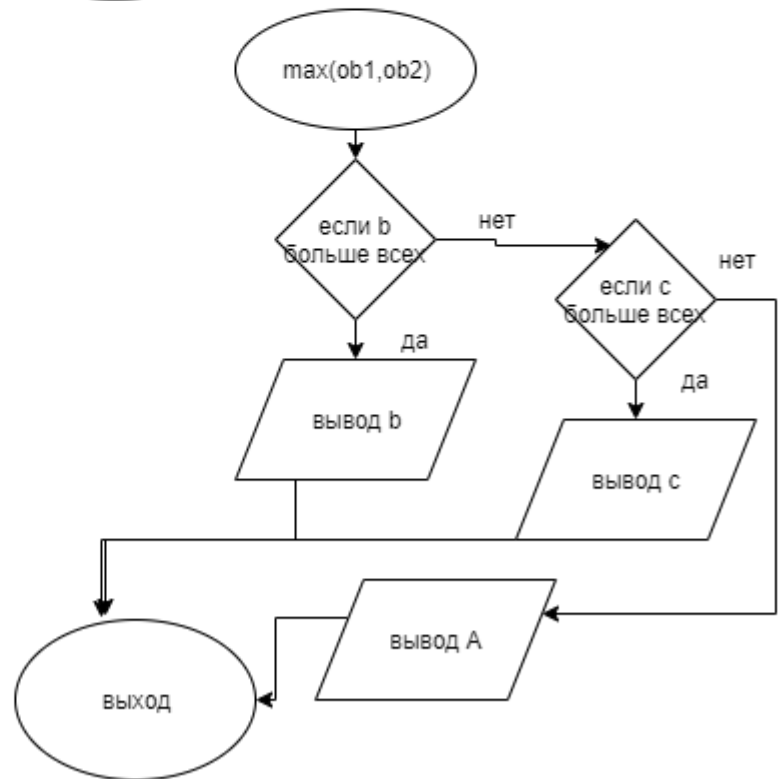
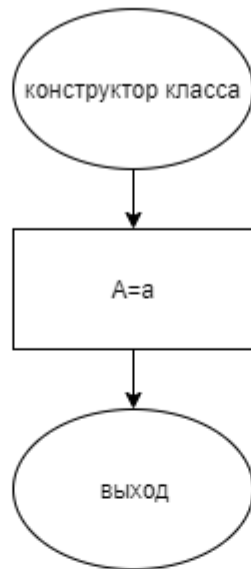
Параметры: нет

Возвращаемое значение: нет

| № | Предикат | Действия | № перехода | Комментарий |
|---|----------|--------------------|------------|-------------|
| 1 | | ввод первого числа | 2 | |

| | | | | |
|---|--|-------------------------------|---|--|
| 2 | | создать объект первого класса | 3 | |
| 3 | | ввод остальных чисел | 4 | |
| 4 | | создать объект второго класса | 5 | |
| 5 | | вызов метода инпут | 6 | |
| 6 | | вызов дружественной функции | ∅ | |

Блок-схема алгоритма



Код программы

Файл cl.cpp

```

#include "cl.h"
#include <iostream>
using namespace std;
cl1::cl1(int a) {
    A = a;
}
void cl2::input(int b, int c) {
    this->b = b;
    this->c = c;
}
int max(cl1 &ob1, cl2 &ob2){
    if (ob2.b > ob2.c && ob2.b > ob1.A) {
        cout << "max = " << ob2.b;
    }
    else if (ob2.c > ob2.b && ob2.c > ob1.A) {
        cout << "max = " << ob2.c;
    }
    else
        cout << "max = " << ob1.A;
    return 0;
}

```

Файл cl.h

```

#ifndef _CL_H
#define _CL_H
class cl2;
class cl1 {
    private:
        int A;
    public:
        cl1(int a);
        friend int max(cl1 &ob1, cl2& ob2);
};
class cl2 {
    private:
        int b, c;
    public:
        void input(int b, int c);
        friend int max(cl1 &ob1, cl2& ob2);
};
#endif

```

Файл main.cpp

```

#include <iostream>
#include "cl.h"
using namespace std;
int main() {
    int a=0, b=0, c=0;
    cin >> a;
    cl1 ob1(a);
    cin >> b >> c;
    cl2 ob2;
    ob2.input(b, c);
    max(ob1, ob2);
}

```

Тестирование

| Входные данные | Ожидаемые выходные данные | Фактические выходные данные |
|----------------|---------------------------|-----------------------------|
| -1 2 0 | max = 2 | max = 2 |
| 5 6 22 | max = 22 | max = 22 |
| 3 5 7 | max = 7 | max = 7 |
| 5 6 11 | max = 11 | max = 11 |

