|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** \_***ИУК «Информатика и управление»*\_\_**\_\_\_\_\_\_\_\_\_\_\_\_

**КАФЕДРА** \_\_***ИУК4 «Программное обеспечение ЭВМ, информационные технологии»***

**ЛАБОРАТОРНАЯ РАБОТА**

**«Классы и объекты в C++»**

**ДИСЦИПЛИНА: «Высокоуровневое программирование»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-22Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Карельский М.К. )  (Подпись) (Ф.И.О.) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Козина А.В. )  (Подпись) (Ф.И.О.) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |
| Калуга , 2021 | | |

**Цель:** приобретение практических навыков и основ объектно-ориентированного программирования, средствами языка C++.

**Задачи:**

1. Изучение основных концепций ООП;
2. Познакомиться с типом данных – «class»;
3. Познакомиться с операторами, предназначенными для работы с классами;
4. Научиться создавать объекты классов;
5. Изучить работу с методами класса;
6. Познакомиться с инициализацией пользовательских объектов.

**Вариант 12**

**Задание:**

1. Определить пользовательский класс:

ЭКЗАМЕН

имя студента – char\*

дата – int

оценка – int

1. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.
2. Определить в классе деструктор.
3. Определить в классе компоненты-функции для просмотра и установки полей данных.
4. Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструктора и деструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор или деструктор вызвал).
5. Определить указатель на экземпляр класса.
6. Определить указатель на компоненту-функцию.
7. Показать в программе использование указателя на объект и указателя на компоненту-функцию.
8. Создать массив объектов, размещенной как в статической области памяти, так и в динамической.
9. Продемонстрировать содержание массивов.

**Блок-схема:**



**Блок-схема 1.** Конструкторы



**Блок-схема 2.** Деструктор



**Блок-схема 3.** Геттеры



**Блок-схема 4.1.** Сеттеры



**Блок-схема 4.2.** Сеттеры



**Блок-схема 5.** Метод вывода данных объекта



**Блок-схема 6.** Демонстрация работы с классом

**Листинг:**

***Exam.h:***

#ifndef EXAM\_H

#define EXAM\_H

class Exam

{

char\* m\_studentName{};

int m\_date{};

int m\_mark{};

public:

Exam();

Exam(const char\* studentName, const int date, const int mark);

Exam(const Exam & exam);

~Exam();

char\* GetStudentName();

int GetDate();

int GetMark();

int SetStudentName(char\* studentName);

int SetDate(int date);

int SetMark(int mark);

void Print();

};

#endif

***Exam.cpp:***

#include "Exam.h"

#include <iostream>

using namespace std;

const int LENGTH\_OF\_STUDENT\_NAME = 255;

Exam::Exam()

{

m\_studentName = new char[LENGTH\_OF\_STUDENT\_NAME]{};

strcpy\_s(m\_studentName, LENGTH\_OF\_STUDENT\_NAME, "Isaac");

m\_date = 10100;

m\_mark = 1;

cout << "\tDefault object is created\n";

Print();

}

Exam::Exam(const char\* studentName, const int date, const int mark)

{

m\_studentName = new char[LENGTH\_OF\_STUDENT\_NAME]{};

strcpy\_s(m\_studentName, LENGTH\_OF\_STUDENT\_NAME, studentName);

m\_date = date;

m\_mark = mark;

cout << "\tCustom object is created\n";

Print();

}

Exam::Exam(const Exam & exam)

{

m\_studentName = new char[LENGTH\_OF\_STUDENT\_NAME]{};

strcpy\_s(m\_studentName, LENGTH\_OF\_STUDENT\_NAME, exam.m\_studentName);

m\_date = exam.m\_date;

m\_mark = exam.m\_mark;

cout << "\tCopied object is created\n";

Print();

}

Exam::~Exam()

{

cout << "\tObject is deleted\n";

Print();

delete m\_studentName;

}

char\* Exam::GetStudentName()

{

return m\_studentName;

}

int Exam::GetDate()

{

return m\_date;

}

int Exam::GetMark()

{

return m\_mark;

}

int Exam::SetStudentName(char\* studentName)

{

if (strlen(studentName) < LENGTH\_OF\_STUDENT\_NAME)

{

strcpy\_s(m\_studentName, LENGTH\_OF\_STUDENT\_NAME, studentName);

return 0;

}

else

{

return -1;

}

}

int Exam::SetDate(int date)

{

if (date < 0)

{

return -1;

}

else

{

int day = date / 10000;

int month = (date % 10000) / 100;

if (day >= 1 && day <= 31 && month >= 1 && month <= 12)

{

m\_date = date;

return 0;

}

else

{

return -1;

}

}

}

int Exam::SetMark(int mark)

{

if (mark >= 1 && mark <= 5)

{

m\_mark = mark;

return 0;

}

else

{

return -1;

}

}

void Exam::Print()

{

cout << "Student name: " << m\_studentName << '\n';

cout << "Date: " << m\_date << '\n';

cout << "Mark: " << m\_mark << '\n';

cout << '\n';

}

***Demo.cpp:***

#include "Exam.h"

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

cout << "\tПример создания и удаления объектов\n\n";

Exam\* defaultIsaac = new Exam();

Exam michaelGeometry = Exam("Michael", 130121, 5);

Exam michaelMath = Exam(michaelGeometry);

michaelMath.SetDate(190121);

michaelMath.Print();

delete defaultIsaac;

cout << "\n\tОпределение указателя на объект\n\n";

Exam\* ivanMath = new Exam("Ivan", 190121, 3);

cout << "\nОпределение указателя на метод класса...\n\n";

void (Exam::\*printPointer) ();

printPointer = &Exam::Print;

cout << "\n\tИспользование указателя на объект для создания копии\n";

cout << "\tИспользование указателя на метод для вызова этого метода\n\n";

Exam \*ivanGeometry = new Exam(\*ivanMath);

ivanGeometry->SetDate(130121);

ivanGeometry->SetMark(4);

(ivanGeometry->\*printPointer)();

cout << "\n\tСоздание статического массива экзаменов Михаила\n";

cout << "\tСоздание динамического массива экзаменов Ивана\n\n";

Exam michaelExams[2]{michaelGeometry, michaelMath};

Exam\* ivanExams = new Exam[2]{\*ivanGeometry, \*ivanMath};

cout << "\n\tВывод массивов\n\n";

for (int i = 0; i < 2; i++)

{

michaelExams[i].Print();

}

for (int i = 0; i < 2; i++)

{

ivanExams[i].Print();

}

cout << "\n\tОчистка кучи\n\n";

delete ivanMath;

delete ivanGeometry;

delete[] ivanExams;

cout << "\n\tОчистка стека\n\n";

return 0;

}

**Тестирование:**

Пример создания и удаления объектов

Default object is created

Student name: Isaac

Date: 10100

Mark: 1

Custom object is created

Student name: Michael

Date: 130121

Mark: 5

Copied object is created

Student name: Michael

Date: 130121

Mark: 5

Student name: Michael

Date: 190121

Mark: 5

Object is deleted

Student name: Isaac

Date: 10100

Mark: 1

Определение указателя на объект

Custom object is created

Student name: Ivan

Date: 190121

Mark: 3

Определение указателя на метод класса...

Использование указателя на объект для создания копии

Использование указателя на метод для вызова этого метода

Copied object is created

Student name: Ivan

Date: 190121

Mark: 3

Student name: Ivan

Date: 130121

Mark: 4

Создание статического массива экзаменов Михаила

Создание динамического массива экзаменов Ивана

Copied object is created

Student name: Michael

Date: 130121

Mark: 5

Copied object is created

Student name: Michael

Date: 190121

Mark: 5

Copied object is created

Student name: Ivan

Date: 130121

Mark: 4

Copied object is created

Student name: Ivan

Date: 190121

Mark: 3

Вывод массивов

Student name: Michael

Date: 130121

Mark: 5

Student name: Michael

Date: 190121

Mark: 5

Student name: Ivan

Date: 130121

Mark: 4

Student name: Ivan

Date: 190121

Mark: 3

Очистка кучи

Object is deleted

Student name: Ivan

Date: 190121

Mark: 3

Object is deleted

Student name: Ivan

Date: 130121

Mark: 4

Object is deleted

Student name: Ivan

Date: 190121

Mark: 3

Object is deleted

Student name: Ivan

Date: 130121

Mark: 4

Очистка стека

Object is deleted

Student name: Michael

Date: 190121

Mark: 5

Object is deleted

Student name: Michael

Date: 130121

Mark: 5

Object is deleted

Student name: Michael

Date: 190121

Mark: 5

Object is deleted

Student name: Michael

Date: 130121

Mark: 5

**Вывод:** в ходе работы были полученые практические навыки создания класса, модификаторов доступа, контрукторов без параметров, с параметрами, конструктора копирования, деструктора, геттеров и сеттеров, объектов класса без параметров, с параметрами, копии объекта, указателей на объект и на метод класса, статического и динамического массивов объектов, удаления объектов и массивов объектов.