

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра ИИТ

Лабораторная работа №4.
"Классы и объекты C++"

Выполнил:
Ст. 2 курса гр. АС-53
Альциванович Н. В
Проверила:
Давидюк Ю. И.

Брест, 2020

1. Цель. Получить практические навыки реализации классов на C++.

2. Постановка задачи (Вариант 1)

Студент

имя – char*

курс – int

пол – bool

Конструкторы:

- Без параметров
- С 3 параметрами
- Копирования

Функции:

- Вывода
- Аллокации памяти

3. Определение пользовательского класса с комментариями.

```
class Student {  
private:  
    char* name;  
    int course;  
    bool sex;  
    void allocateChars(const char* _name);  
  
public:  
    Student();  
    Student(const char* _name, int _course, bool _sex);  
    Student(const Student&);  
    ~Student();  
    void print();  
};
```

4. Реализация конструкторов и деструктора и других функций

```
void Student::allocateChars(const char* _name) { // Выделяем память  
    name = new char[strlen(_name) + 1];  
    strcpy(name, _name);  
}  
  
Student::Student() { // Конструктор без параметров  
    allocateChars("Unnamed");  
    std::cout << "Create class Student without constructor params." << std::endl;  
}  
  
Student::Student(const char* _name, int _course, bool _sex) { // Конструктор с 3 параметрами  
    allocateChars(_name);  
    course = _course;  
    sex = _sex;  
    std::cout << "Create class Student with 3 constructor params." << std::endl;  
}  
  
Student::Student(const Student& _Student) { // Конструктор копирования  
    allocateChars(_Student.name);  
    course = _Student.course;  
    sex = _Student.sex;  
    std::cout << "Create class Student with copy constructor." << std::endl;  
}
```

```

Student::~~Student() { // Деструктор
    std::cout << "Call Student class \"" << name << "\" desctructor." << std::endl;
    delete[] name;
}

void Student::print() { // Вывод
    std::cout << "\nName: " << name << "\nCourse: " << course
        << "\nSex: "; if (sex) std::cout << "female"; else std::cout << "male";
    std::cout << "\n" << std::endl;
}

```

5. Фрагмент программы, показывающий использование указателя на объект с объяснением.

`Student* Student1 = new Student("Nikolay", 2, 0);` – создаем объект класса с именем `Student1` параметрами «Nikolay», 2, 0 и получаем на него указатель.

6. Листинг основной программы, в котором должно быть указано, в каком месте и какой конструктор или деструктор вызываются.

```

int main() {
    Student* Student1 = new Student("Nikolay", 2, 0);
    Student Student2;
    Student Student3(*Student1);

    Student1->print();
    Student2.print();
    Student3.print();

    delete(Student1);

    return 0;
}

```

```

Create class Student with 3 constructor params.
Create class Student without constructor params.
Create class Student with copy constructor.

Name: Nikolay
Course: 2
Sex: male

Student() { // Конструктор копирования
Name: Unnamed
Course: 0
Sex: male
Student with copy constructor." << std::endl;

Name: Nikolay
Course: 2
Sex: male
Student::~~Student() { // Деструктор
    std::cout << "Call Student class \"" << name << "\" desctructor." << std::endl;
    delete[] name;
}

Call Student class "Nikolay" desctructor.
Call Student class "Unnamed" desctructor.
Call Student class "Nikolay" desctructor.

```

7. Вывод:

Получил практические навыки реализации классов на C++.