

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Институт естественных и точных наук
Кафедра прикладной математики и программирования

ОТЧЕТ

о выполнении лабораторной работы № 1 по дисциплине
«Объектно-ориентированное программирование»

Автор работы,
студент группы ЕТ-211
_____ Савонин М.В.
« ____ » _____ 2022 г.

Руководитель работы,
старший преподаватель
_____ Демидов А.К.
« ____ » _____ 2022 г.

Челябинск 2022

1 ПОСТАНОВКА ЗАДАЧИ

I. Реализовать класс

```
struct Data
{
    long prioritet;
    string info;
};
class PQueue
{
public:
    PQueue(int maxsize=100);
    ~PQueue(){ delete[] data;}
    void add(const Data &dat); //добавить данные
    const Data& first(); //получить данные с наименьшим приоритетом
    void next(); //убрать данные с наименьшим приоритетом
    bool isEmpty(); //очередь пуста
    bool isFull(); //очередь полна
    void print(); //распечатать состояние объекта
};
```

II. Реализовать main с тестами для проверки класса (создание объекта и выполнение действий с ним)

2 ОПИСАНИЕ ИНТЕРФЕЙСА КЛАССА

```
struct Data
{
    long prioritet;
    string info;
};
class PQueue
{
    int n; //количество элементов
    int size; //максимальное количество элементов
    int firstNum; //первый элемент
    Data *data; //массив очереди
public:
    PQueue(int maxsize=100);
    ~PQueue(){ delete[] data;}
    void add(const Data &dat); //добавить данные
    const Data& first(); //получить данные с наименьшим приоритетом
    void next(); //убрать данные с наименьшим приоритетом
    bool isEmpty(); //очередь пуста
    bool isFull(); //очередь полна
    void print(); //распечатать состояние объекта
};
```

3 ОПИСАНИЕ ТЕСТОВ ДЛЯ ПРОВЕРКИ КЛАССОВ

```
PQueue a = PQueue(10);
Data b;
for(int i = 0; i < 10; i++)
{
    b.info = 'a'+i;
    b.prioritet = i;
    a.add(b);
}
b = a.first();
cout << b.info << ' ' << b.prioritet << endl; // a 0
a.print(); \* 10 10
      a b c d e f g h i j
      0 1 2 3 4 5 6 7 8 9*/
a.next();
a.print(); \* 9 10
      b c d e f g h i j
      1 2 3 4 5 6 7 8 9*/
cout << a.isFull() << ' ' << a.isEmpty(); // 0 0
```

4 ЛИСТИНГ РЕАЛИЗАЦИИ КЛАССА

```
PQueue::PQueue(int maxsize)
{
    size = maxsize;
    n = 0;
    firstNum = 0;
    data = new Data[maxsize];
}

void PQueue::add(const Data &dat)
{
    if(isFull())
    {
        cout << "Очередь переполнена" << endl;
        exit(1);
    }
    data[firstNum+n] = dat;
    n++;
}

const Data& PQueue::first()
{
    Data& a = data[firstNum];
    for(int i = 1; i < n; i++)
    {
        if(a.prioritet > data[firstNum+i].prioritet) a = data[firstNum+i];
    }
    return a;
}

void PQueue::next()
{
    Data& a = data[firstNum];
    int k=firstNum;
    for(int i = 1; i < n; i++)
    {
        if(a.prioritet > data[firstNum+i].prioritet)
        {
            a = data[firstNum+i];
            k = i;
        }
    }
    for(int i = k+firstNum; i > firstNum; i--)
    {
        data[i] = data[i-1];
    }
    n--;
    firstNum++;
}
```

```

bool PQueue::isEmpty()
{
    return n == 0;
}

bool PQueue::isFull()
{
    return n == size;
}

void PQueue::print()
{
    cout << n << ' ' << size << endl;
    for(int i = 0; i < n; i++)
    {
        cout << data[firstNum+i].info << ' ';
    }
    cout << endl;
    for(int i = 0; i < n; i++)
    {
        cout << data[firstNum+i].prioritet << ' ';
    }
}

```