

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЯ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №4
по дисциплине Объектно-Оrientированные Технологии Программирования и Стандарты
Проектирования за I семестр
Тема: "Перегрузка операций"

Выполнил:
студент 2 курса
IV семестра
факультета ЭИС
группы ПО-4(1)
Галанин П. И.
«__» _____ 2021 г.

Проверил:
магистрант
кафедры ИИТ
Миндер А. В.
«__» _____ 2021 г.

Отчёт по лабораторной работе №4

Тема: «Перегрузка операций»

Цель работы: Получить практические навыки создания абстрактных типов данных и перегрузки операций в языке C++.

Ход работы:

Порядок выполнения работы.

1. Выбрать класс абстрактного типа данных (АТД) в соответствии с вариантом.
2. Определить и реализовать в классе конструкторы, деструктор, функции Input (ввод с клавиатуры) и Print (вывод на экран), перегрузить операцию присваивания.
3. Написать программу тестирования класса и выполнить тестирование.
4. Дополнить определение класса заданными перегруженными операциями (в соответствии с вариантом).
5. Реализовать эти операции. Выполнить тестирование.

Вариант 5:

АТД - множество с элементами типа char. Дополнительно перегрузить следующие операции:

- () - конструктор множества (в стиле конструктора Паскаля);
- + - объединение множеств;
- <= - сравнение множеств.

Листинг: main.cpp

```
1 #include <iostream>
2 using namespace std;
3
4 class myCharSet
5 {
6 private:
7     char* arr;
8     int length;
9 public:
10    myCharSet();
11    ~myCharSet();
12    void add(char ch);
13    void print();
```

					ЛР.190333.ПО4.04 81 00		
Изм.	Лист	№ докум.	Подп.	Дата	Отчёт по лабораторной работе №4 Перегрузка операций		
Разраб.	Галанин						
Пров.	Миндер						
Н. контр.							
Утв.							
					Лит.	Лист	Листов
					Л	2	5
					БрГТУ		

```

14     friend myCharSet operator+(const myCharSet& obj1, const myCharSet& obj2);
15     friend bool operator<=(const myCharSet &obj1, const myCharSet &obj2);
16 };
17
18 int main()
19 {
20     myCharSet x1 = myCharSet();
21     cout << "x1" << endl;
22     x1.add('c');
23     x1.add('a');
24     x1.add('b');
25     x1.add('b');
26     x1.add('c');
27     x1.add('c');
28     x1.print();
29
30     myCharSet x2 = myCharSet();
31     cout << "x2" << endl;
32     x1.add('c');
33     x2.add('a');
34     x2.add('d');
35     x2.add('c');
36     x2.add('d');
37     x2.print();
38
39     cout << "(x1 <= x2) = " << (x1 <= x2) << endl << endl;
40     cout << "(x2 <= x1) = " << (x2 <= x1) << endl << endl;
41     cout << "(x1 <= x1) = " << (x1 <= x1) << endl << endl;
42
43     cout << "x3" << endl;
44     myCharSet x3 = myCharSet();
45     x3.add('g');
46     x3.add('d');
47     x3.add('u');
48     x3.add('w');
49     x3.print();
50
51     cout << "x4" << endl;
52     myCharSet x4 = myCharSet();
53     x4.add('t');
54     x4.add('l');
55     x4.add('s');
56     x4.add('d');
57     x4.print();
58
59     cout << "x5 = x3 + x4" << endl;
60     myCharSet x5 = x3 + x4;
61     x5.print();
62
63     return 0;
64 }
65
66 myCharSet::myCharSet()
67 {
68     cout << this << " constructor" << endl;
69     this->length = 0;
70     char* arr = new char [this->length];
71     this->arr = arr;
72 }
73

```

```

74 myCharSet::~myCharSet()
75 {
76     cout << this << " destructor" << endl;
77     delete [] arr;
78 }
79
80 void myCharSet::add(char ch)
81 {
82     int k = 0;
83     for (int i = 0; i < this->length; i++)
84     {
85         if (ch == this->arr[i])
86         {
87             k += 1;
88         }
89     }
90
91     if (k == 0)
92     {
93         this->length += 1;
94         char* newArr = new char [this->length];
95         newArr[this->length - 1] = ch;
96         for (int i = 0; i < this->length; i++)
97         {
98             newArr[i] = this->arr[i];
99         }
100        delete [] this->arr;
101        newArr[this->length - 1] = ch;
102        this->arr = newArr;
103    }
104
105    for (int i = 0; i < this->length; i++)
106    {
107        for (int j = 0; j < this->length; j++)
108        {
109            if (this->arr[i] < this->arr[j])
110            {
111                int temp = this->arr[i];
112                this->arr[i] = this->arr[j];
113                this->arr[j] = temp;
114            }
115        }
116    }
117 }
118
119 void myCharSet::print()
120 {
121     printf("[ ");
122     for (int i = 0; i < this->length; i++)
123     {
124         printf("%c, ", this->arr[i]);
125     }
126     printf("]\n\n");
127 }
128
129 bool operator <=(const myCharSet &obj1, const myCharSet &obj2)
130 {
131     int length = min(obj1.length, obj2.length);
132
133     for (int i = 0; i < length; i++)

```

```

134     {
135         if (obj1.arr[i] > obj2.arr[i])
136         {
137             return false;
138         }
139     }
140
141     return true;
142 }
143
144 myCharSet operator+(const myCharSet& obj1, const myCharSet& obj2)
145 {
146     myCharSet obj = myCharSet();
147
148     for (int i = 0; i < obj1.length; i++)
149     {
150         obj.add(obj1.arr[i]);
151     }
152     for (int i = 0; i < obj2.length; i++)
153     {
154         obj.add(obj2.arr[i]);
155     }
156
157     return obj;
158 }

```

Листинг: Out

```

1 0x7ffa24d6780 constructor
2 x1
3 [ a, b, c, ]
4
5 0x7ffa24d6770 constructor
6 x2
7 [ a, c, d, ]
8
9 (x1 <= x2) = 1
10
11 (x2 <= x1) = 0
12
13 (x1 <= x1) = 1
14
15 x3
16 0x7ffa24d6760 constructor
17 [ d, g, u, w, ]
18
19 x4
20 0x7ffa24d6750 constructor
21 [ d, l, s, t, ]
22
23 x5 = x3 + x4
24 0x7ffa24d6740 constructor
25 [ d, g, l, s, t, u, w, ]
26
27 0x7ffa24d6740 destructor
28 0x7ffa24d6750 destructor
29 0x7ffa24d6760 destructor
30 0x7ffa24d6770 destructor
31 0x7ffa24d6780 destructor

```

Изм	Лист	№ докум.	Подп.	Дата

ЛР.190333.ПО4.04 81 00

Лист

5