

Практическая работа №1

Тема: Технология реализации алгоритмов с использованием функции.

Постановка задачи: разработать программу выполнения операций над статическим массивом целых чисел.

1) Функция ввода элементов массива заданного размера.

```
8   int InputArray(int* arr, int size) //Функция ввода массива
9   {
10  for (int i = 0; i < size; i++)
11  {
12      cout << "Введите число №" << i + 1 << "->";
13      cin >> arr[i];
14      cout << endl;
15  }
16  return *arr;
17 }
```

2) Функция вывода массива заданного размера.

```
20 void OutputArray(int* arr, int size) //Функция вывода массива
21 {
22     cout << "OutputArray: \n";
23     for (int i = 0; i < size; i++)
24     {
25         cout << arr[i] << " ";
26     }
27     cout << endl;
28 }
```

3) Функция заполнения массива случайными числами.

```
31 void RArray(int* arr, int size) //Функция заполнения массива случайными значениями
32 {
33     cout << "RArray: \n";
34     for (int i = 0; i < size; i++)
35     {
36         arr[i] = -30 + rand() % 100;
37         cout << arr[i] << " ";
38     }
39     cout << endl;
40 }
```

4) Функция поиска первого вхождения значения в массив.

```
43 void FDigit(int* arr, int size, int val) //Функция нахождения первого вхождения значения
44 {
45     int based = 0;
46     int in = 0;
47     for (; in < size; in++)
48     {
49         if (arr[in] == val)
50         {
51             based = 1;
52             break;
53         }
54     }
55     if (based)
56     {
57         cout << in << " ";
58     }
59     else
60     {
61         cout << "-1" << " ";
62     }
63 }
64 }
```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	43	1
2	Размер: 3 Массив: -6 -43 53	-6	0
3	Размер: 5 Массив: 0 0 1 0 0	0	0
4	Размер: 6 Массив: 5 5 6 6 3 5	6	2
5	Размер: 7 Массив: 5 43 53 23 3 4 0	3	4

5) Функция нахождения индекса первого отрицательного числа в массиве.

```

67 void Fminus(int* arr, int size) //Функция первого вхождения отрицательного значения
68 {
69     int based = 0;
70     int in = 0;
71     for (; in < size; in++)
72     {
73         if (arr[in] < 0)
74         {
75             based = 1;
76             break;
77         }
78     }
79
80     if (based)
81     {
82         cout << in << " ";
83     }
84     else
85     {
86         cout << "-1" << " ";
87     }
88 }

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	-	-5
2	Размер: 3 Массив: -6 -43 53	-	-6
3	Размер: 5 Массив: 0 0 1 0 0	-	-1
4	Размер: 6 Массив: 5 5 6 6 3 5	-	-1
5	Размер: 7 Массив: 5 43 53 23 3 4 0	-	-1

б) Функция поиска всех вхождений значения.

```

91 void FindDigits(int* arr, int size, int val) //Функция нахождения всех вхождений значения
92 {
93     int a = 0;
94     for (int i = 0; i < size; i++)
95     {
96         if (arr[i] == val)
97         {
98             cout << i << " ";
99             a++;
100         }
101     }
102     if (a == 0)
103     {
104         cout << "-1";
105     }
106     cout << endl;
107 }
108

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	43	1 2
2	Размер: 3 Массив: -6 -43 53	53	2
3	Размер: 5 Массив: 0 0 1 0 0	0	0 1 3 4
4	Размер: 6 Массив: 5 5 6 6 3 5	6	2 3
5	Размер: 7 Массив: 5 43 53 23 3 4 0	4	5

7) Функция вставки нового значения в заданную позицию массива.

```

111 void IValue(int* arr, int size, int idx, int val) //Функция замены элемента на новый
112
113 {
114     arr[idx] = val;
115     for (int i = 0; i < size; i++)
116     {
117         cout << arr[i] << " ";
118     }
119     cout << endl;
120 }

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	0 0	0 43 43 35 -5
2	Размер: 3 Массив: -6 -43 53	2 101	-6 -43 101
3	Размер: 5 Массив: 0 0 1 0 0	2 0	0 0 0 0 0
4	Размер: 6 Массив: 5 5 6 6 3 5	3 55	5 5 6 55 3 5
5	Размер: 7 Массив: 5 43 53 23 3 4 0	4 111	5 43 53 23 111 4 0

8) Функция алгоритма удаления со сжатием из массива значения в заданной позиции, сохраняя порядок следования остальных элементов.

```

123 void RValue(int* arr, int* size, int idx) //Функция удаления элемента в заданной позиции
124 {
125     {
126     for (int i = idx; i < *size - 1; i++)
127     {
128         arr[i] = arr[i + 1];
129     }
130     (*size)--;
131     }
132     for (int j = 0; j < *size; j++)
133     {
134         cout << arr[j] << " ";
135     }
136     cout << endl;
137 }

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	2	0 43 35 -5
2	Размер: 3 Массив: -6 -43 53	2	-6 101
3	Размер: 5 Массив: 0 0 1 0 0	3	0 0 0 0
4	Размер: 6 Массив: 5 5 6 6 3 5	1	5 6 55 3 5
5	Размер: 7 Массив: 5 43 53 23 3 4 0	4	5 43 53 23 4 0

9) Функция алгоритма удаления со сжатием всех вхождений заданного значения из массива. Сложность алгоритма $O(n^2)$.

```

140 void R0ccurences1(int* arr, int* size, int key) //  $\theta(n^2)$ 
141 {
142     for (int i = 0; i < *size; i++)
143     {
144         if (arr[i] == key)
145         {
146             for (int j = i; j < *size - 1; j++)
147             {
148                 arr[j] = arr[j + 1];
149             }
150             (*size)--;
151         }
152         else
153         {
154             i++;
155         }
156     }
157     for (int k = 0; k < *size; k++)
158     {
159         cout << arr[k] << " ";
160     }
161 }
162

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	43	0 35 -5
2	Размер: 3 Массив: -6 -43 53	-6	101

3	Размер: 5 Массив: 0 0 1 0 0	0	
4	Размер: 6 Массив: 5 5 6 6 3 5	5	6 55 3
5	Размер: 7 Массив: 5 43 53 23 3 4 0	43	5 53 23 4 0

10) Функция алгоритма удаления со сжатием всех вхождений заданного значения из массива. Сложность алгоритма $O(n)$.

```

165 void R0Ccurences(int* arr, int* size, int key) //O(n)
166
167 {
168     int j = 0;
169     for (int i = 0; i < *size; i++)
170     {
171         arr[j] = arr[i];
172         if (arr[i] != key)
173             j++;
174     }
175     *size = j;
176     for (int k = 0; k < *size; k++)
177     {
178         cout << arr[k] << " ";
179     }
180     cout << endl;
181 }

```

Таблица тестов функции:

1	Размер: 5 Массив: 45 43 43 35 -5	35	0 -5
2	Размер: 3 Массив: -6 -43 53	101	
3	Размер: 5 Массив: 0 0 1 0 0	0	
4	Размер: 6 Массив: 5 5 6 6 3 5	55	6 3
5	Размер: 7 Массив: 5 43 53 23 3 4 0	23	5 53 4 0

11) Реализация функций.

```
162 int main()
163 {
164     srand(time(0));
165     setlocale(LC_ALL, "Russian");
166     cout << "Ввести размер массива? (y/N) " << endl;
167     char ch = 'n';
168     int n = 0;
169     cin >> ch;
170     if (ch == '\n' || tolower(ch) == 'n')
171     {
172         cout << "Размер массива -> 1000" << endl;
173         n = 1000;
174     }
175     else
176     {
177         cout << "Введите размер массива -> ";
178         // Задаем размер массива
179         cin >> n;
180     }
181     int* mas = (int*)malloc(n * sizeof(int));
182
183     // Функция ввода
184     InputArray(mas, n);
185     // Функция вывода
186     OutputArray(mas, n);
187     // Функция заполнения массива случайными значениями
188     RandomArray(mas, n);
189
190     cout << endl << "Введите новые значения массива ->" << endl;
191     InputArray(mas, n);
192
193     cout << "FirstDigit: " << endl;
194     int cal = 0;
195     cout << "Введите значение -> ";
196     cin >> cal;
197     // Функция нахождения первого вхождения значения
198     FirstDigit(mas, n, cal);
199
200     cout << endl;
201
202     cout << "FirstNegative: " << endl;
203     // Функция первого вхождения отрицательного значения
204     FirstNegative(mas, n);
205
206     cout << endl;
207
208     cout << "FindDigits: " << endl;
209     cout << "Введите значение, вхождения которого нужно найти ->" << endl;
210     cin >> cal;
211     // Функция нахождения всех вхождений значения
212     FindDigits(mas, n, &cal);
```

```

213
214     cout << endl;
215
216     int* mas2 = new int[n];
217     mas2 = mas;
218
219     cout << "FindDigits: " << endl;
220     cout << "Введите индекс и новое значение ->" << endl;
221     int index;
222     cin >> index;
223     cin >> cal;
224     // Функция замены элемента на новый
225     InsertValue(&mas2, &n, index, cal);
226
227     cout << "RemoveValue: " << endl;
228     cout << "Введите индекс ->" << endl;
229     cin >> index;
230     RemoveValue(&mas2, &n, index);
231
232     cout << "RemoveOccurences1: " << endl;
233     cout << "Введите значение ->" << endl;
234     cin >> cal;
235     RemoveOccurences2(&mas2, &n, cal);
236
237     cout << "RemoveOccurences2: " << endl;
238     cout << "Введите значение ->" << endl;
239     cin >> cal;
240     RemoveOccurences(&mas2, &n, cal);
241 }

```