

**"Федеральное государственное автономное образовательное учреждение высшего образования
"Национальный исследовательский университет "Высшая школа экономики"**

Московский институт электроники и математики им. А.Н. Тихонова НИУ ВШЭ Департамент компьютерной инженерии (или департамент электронной инженерии)

Раздел	Макс. оценка	Итог., оценка	Итог., оценка	Итог., оценка
Постановка	0,5			
Метод	1			
Спецификация	0,5			
Алгоритм	1,5			
Работа программы	1			
Листинг	0,5			
Тесты	1			
Вопросы	2			
Доп. задание	2			
Итого	10			

ОТЧЕТ
по лабораторной работе №3
Студент:
Пырлицану Никита
Евгеньевич
Группа: БИВ239
Вариант: №473 (3, 5, 4)
Руководитель:
Батонова Оксана Юрьевна
Оценка: _____
Дата сдачи: _____

Содержание

Задания	1
Постановка задачи	2
Метод решения	3
Задание I.	3
Задание II.	3
Задание III.	3
Внешняя спецификация	4
Задание I	4
Задание II	4
Задание III	4
Описание алгоритма на псевдокоде	5
Задание I	5
Задание II	5
Задание III	6
Листинг программ	7
Задание I	7
Задание II	7
Задание III	8
Тесты	12

Задания

I. Функция $y = f(x)$ задана таблицей:

X_i	X_1	X_2	X_3
Y_i	Y_1	Y_2	Y_3

Вычислить приближенное значение в точке по формуле:

$$f(x) = \begin{cases} y_1 + \frac{x-x_1}{x_2-x_1}(y_2-y_1), & \text{если } x_1 \leq x < x_2 \\ y_2 + \frac{x-x_2}{x_3-x_1}(y_3-y_2), & \text{если } x_2 \leq x < x_3 \\ y_3, & \text{если } x = x_3 \end{cases}$$

II. Вычислить значение по формуле:

$$H = \min_{j=1,m} \max_{i=1,n} |b_{i,j}|, \text{ где } b_{i,j} - \text{элементы матрицы } B[0:n-1, 0:m-1]$$

III. Дан целочисленный массив $A[1:n]$. Написать программу, включающую две функции с параметрами. В первой функции необходимо подсчитать количество повторений каждого элемента массива A . Вторая процедура решает следующую задачу: удалить из массива A все неповторяющиеся элементы путем сдвига (дополнительный массив не использовать).

Постановка задачи

Дано

- I. $x_1, x_2, x_3, y_1, y_2, y_3$ - вещ., x - вещ.
- II. $lmax$ - нат., n, m - нат., $B[0:n-1, 0:m-1]$ - цел.
- III. $lmax$ - нат., n - нат., $A[0:n-1]$ - цел.

Результат

- I. $f(x)$ - вещ.
- II. H - цел.
- III. $A[0:k-1]$ - цел. или <Nothing to delete!>

При

- I. $x_1 \leq x, x_2 \leq x_3$
- II. $n, m \geq 1$ и $n, m < lmax$
- III. $n \geq 1$ и $n \leq lmax$

Связь

I. См. формулу в задании

II. См. формулу в задании

III. k - длина массива A' (A после удаления повторов)

$$\forall i = \overline{0, n-1} : \exists l = \overline{0, n-1} : i \neq l \text{ и } A[i] = A[l] \quad \exists j = \overline{0, k-1} : A[i] = A'[j]$$

Метод решения

Задание I.

Метод функции f:

вернуть $y_1 + \frac{x-x_1}{x_2-x_1}(y_2-y_1)$, если $x_1 \leq x < x_2$
вернуть $y_2 + \frac{x-x_2}{x_3-x_2}(y_3-y_2)$, если $x_2 \leq x < x_3$
вернуть y_3 , если $x = x_3$

Задание II.

Метод функции H:

$min = MAX_NUMBER$
 $max = 0$
$$\left\{ \begin{array}{l} \text{для } j = \overline{0, m-1} \\ \left\{ \begin{array}{l} \text{для } i = \overline{0, m-1} \\ max = |B[i, j]|, \text{ если } |B[i, j]| > max \\ min = max, \text{ если } max > min \end{array} \right. \end{array} \right.$$

вернуть min

Задание III.

Метод функции findRep:

$bn = 0$
 $j = 0$
$$\left\{ \begin{array}{l} \text{для } i = \overline{0, n-1} \\ \left\{ \begin{array}{l} \text{пока } j < bn \text{ и } reps[j] \neq arr[i] \\ j = j + 1 \end{array} \right. \\ reps[bn] = A[i]; reps[bn] = 1; bn = bn + 1, \text{ если } j = bn \\ reps[j] = reps[j] + 1 \text{ в противном случае} \end{array} \right.$$

Метод функции delNonRep:

$j = 1$
$$\left\{ \begin{array}{l} \text{для } i = \overline{0, bn-1} \\ \left\{ \begin{array}{l} \text{пока } a[j] \neq reps[i] \\ j = j + 1 \end{array} \right. , \text{ если } reps[i] = 1 \\ \left\{ \begin{array}{l} \text{для } l = j + 1 \\ a[l-1] = a[l] \\ k = k - 1 \end{array} \right. \end{array} \right.$$

Метод решения:

$bn = findRep()$
 $delNonRep()$

Внешняя спецификация

Задание I

Task I Enter $x_1, x_2, x_3, y_1, y_2, y_3$ such that $x_1 \leq x_2 \leq x_3$
--

$$\left\{ \boxed{\langle x_1 \rangle \langle x_2 \rangle \langle x_3 \rangle \langle y_1 \rangle \langle y_2 \rangle \langle y_3 \rangle} \right\}^{\star} \text{ до } x_1 \leq x_2 \leq x_3$$

$$\left\{ \begin{array}{l} \boxed{\text{Enter } x \text{ such that } x_1 \leq x \leq x_3 :} \\ \left\{ \boxed{\langle x \rangle} \right\}^{\star} \text{ до } x_1 \leq x \leq x_3 \\ \boxed{f(\ll x \gg) = \ll f(x) \gg} \end{array} \right\}^{\star} \text{ до } 1 \neq 1$$

Задание II

Task II. Enter n, m such that $1 \leq n, m \leq \ll lmax \gg$
--

$$\left\{ \boxed{\langle n \rangle \langle m \rangle} \right\}^{\star} \text{ до } 1 \leq n, m \leq lmax$$

Enter matrix $\ll n \gg \times \ll m \gg$: Строка 1: $\langle B[1, 1] \rangle \langle B[1, 2] \rangle \dots \langle B[1 : m] \rangle$... Строка n : $\langle B[n, 1] \rangle \langle B[n, 2] \rangle \dots \langle B[n, m] \rangle$ $H = \ll H() \gg$

Задание III

Enter n greater than 1 and less than $\ll lmax \gg$:

$$\left\{ \boxed{\langle n \rangle} \right\}^{\star} \text{ до } 1 \leq n \leq lmax$$

<i>Enter array with $\ll n \gg$ elements :</i> $\langle A[1] \rangle \dots \langle A[n] \rangle$ $A = \{ \ll A[1] \gg, \dots, \ll A[k] \gg \}$

Описание алгоритма на псевдокоде

Задание I

Алг «Функция f»

Вход $x_1, x_2, x_3, y_1, y_2, y_3, x -$.

Выход $ret -$.

нач

если $x_1 \leq x$ и $x < x_2$ то
 $ret := y_1 + \frac{x-x_1}{x_2-x_1}(y_2 - y_1)$

все если $x_2 \leq x$ и $x < x_3$ то
 $ret := y_2 + \frac{x-x_2}{x_3-x_2}(y_3 - y_2)$

все $ret := y_3$

кон

Алг «Лабораторная работа №3. Задание I»

нач

вывод («Task I»)

вывод (<< Enter $x_1, x_2, x_3, y_1, y_2, y_3$ >>)

цикл

ввод ($x_1, x_2, x_3, y_1, y_2, y_3$)

до $x_1 \leq x_2 \leq x_3$

цикл

ввод (x)

до $x_1 \leq x \leq x_3$

вывод («f(», x , «=», $f(x_1, x_2, x_3, y_1, y_2, y_3, x)$)

кон

Задание II

Алг «Функция H»

Вход: n, m, B

Выход: min

нач

$min := MAX_NUMBER$

цикл от $j = 0$ до $m - 1$

$max := B[0, j]$

цикл от $i = 0$ до $n - 1$

если $|B[i, j]| > |max|$ то

$max := B[i, j]$

все

кц если $|max| < |min|$ то

$min = max$ // все

кц

кон

Алг «Лабораторная работа №3. Задание II»

нач

вывод («Task II»)

вывод («Enter n, m such that $1 \leq n, m \leq lmax$ »)

цикл

ввод (n, m)

до $1 \leq n \leq lmax$ и $1 \leq m \leq lmax$

вывод («Enter matrix », n , «x», m)

ввод ($B[0 : n - 1]$)

вывод («H = », $H(n, m, b)$)

кон

Задание III**Алг** «Функция findRep»**Вход:** $n, arr, reps, repc$ **Выход:** bn **нач** $bn := 0$ **цикл** **от** $i = 0$ **до** $n - 1$ $j := 0$ **цикл-пока** $j < bn$ **и** $reps[j] \neq arr[i]$ $j := j + 1$ **кц****если** $j = bn$ **то** $reps[bn] := arr[i]$ $repc[bn] := 1$ $bn := bn + 1$ **иначе** $repc[j] := repc[j] + 1$ **все****кон****Алг** «Функция delNonRep»**Вход:** $n, arr, cl_arr_len, reps, repc$ **Выход:** n **нач** $j := 0$ **цикл** **от** $i = 0$ **до** $bn - 1$ **если** $repc[i] = 1$ **то****цикл-пока** $a[j] \neq reps[i]$ $j := j + 1$ **кц****цикл** **от** $l = j + 1$ **до** $k - 1$ $a[l - 1] := a[l]$ **кц** $k := k - 1$ **все****кон****Алг** «Лабораторная работа №3. Задание III»**нач****вывод** («Task III.»)**вывод** («Enter n greater than 0 and less then», lmax) $k := n$ **вывод** («Enter array with», n, «elements:») $bn := findRep(n, a, reps, repc)$ **вывод** («A = [», A[0], A[1], ... A[k-1], «]»)**кон**

Листинг программ

Задание I

```

1 #include <stdio.h>
2 #include "../..../safeIO.h"
3
4 float f(float x1, float x2, float x3, float y1, float y2, float y3, float x) {
5     if (x1 <= x && x < x2) {
6         return (float)y1 + ((float)(x-x1) / (x2-x1))*(y2-y1);
7     }
8
9     if (x2 <= x && x < x3) {
10        return (float)y2 + ((float)(x-x2) / (x3-x2))*(y3-y2);
11    }
12
13    return y3;
14 }
15
16 int main() {
17     printf("Task I.\n");
18     float x1, x2, x3, y1, y2, y3, x;
19
20
21     do {
22         printf("Enter x1, x2, x3, y1, y2, y3:\n");
23         enterFloats(6, &x1, &x2, &x3, &y1, &y2, &y3);
24     } while (!(x1 <= x2 && x2 <= x3));
25
26     for (;;) {
27         printf("Enter x such that x1 <= x <= x3:\n");
28         enterFloatRanged(&x, x1, x3);
29         printf("f(%f) = %f\n", x, f(x1,x2,x3,y1,y2,y3, x));
30     }
31
32
33     return 0;
34 }

```

Задание II

```

1 ##include <stdio.h>
2 #include <stdlib.h>
3 #include "../..../safeIO.h"
4 #include <limits.h>
5
6 #define lmax 10
7
8 int H(int n, int m, int b[][lmax]) {
9     int max, min = INT_MAX;
10
11     for (int *uj=*b; uj<(*b)+m; uj++) {
12         max = *uj;
13         for (int *ui=uj+lmax; ui<b[n-1]+m; ui+=lmax)
14             if (abs(*ui) > max)
15                 max = abs(*ui);
16
17         if (max < min)
18             min = max;
19     }
20
21     return min;
22 }

```

```

23
24 int main() {
25     printf("Task II.\n");
26     int b[lmax][lmax];
27     int n, m;
28
29     printf("Enter n, m such that 1 <= n, m <= %d:\n", lmax);
30     enterIntsRanged(1, lmax, 2, &n, &m);
31
32     printf("Enter matrix %dx%d:\n", n, m);
33     for (int i=0; i<n; i++) {
34         printf("String %d:\n", i+1);
35         enterIntArray(b[i], m);
36     }
37
38     printf("H = %d", H(n, m, b));
39 }

```

Задание III

```

1 #include <stdio.h>
2 #include "../safeIO.h"
3 #define lmax 10
4
5 int findRep(int n, int *arr, int* reps, int* repc) {
6     int *bn=reps;
7     int *j;
8     for (int *i=arr; i<arr+n; i++) {
9         for (j=reps; j<bn && *j != *i; j++);
10
11         if (j == bn) {
12             *bn = *i;
13             *(repc-reps+bn) = 1;
14             bn++;
15         }
16         else {
17             (*(repc-reps+j))++;
18         }
19     }
20
21     return (int)(bn-reps);
22 }
23
24 int delNonRep(int n, int *a, int cl_arr_len, int *reps, int *repc) {
25     int *j=a;
26     for (int *i=reps; i<reps+cl_arr_len; i++) {
27         if (*(repc-reps+i) == 1) {
28             while(*j != *i) j++;
29             for (int *l=j+1; l<a+n; l++) {
30                 *(l-1) = *l;
31             }
32             n--;
33         }
34     }
35
36     return n;
37 }
38
39
40 int main() {
41     printf("Task III.\n");
42     int a[lmax], reps[lmax], repc[lmax];
43     int n;
44 }

```

```

45     printf("Enter n greater than 0 and less then %d:\n", lmax);
46     enterIntRanged(&n, 1, lmax);
47
48     printf("Enter array with %d elements:\n", n);
49     enterIntArray(a, n);
50
51     int cl_arr_len = findRep(n, a, reps, repc);
52     int k = delNonRep(n, a, cl_arr_len, reps, repc);
53
54     if (k == n) {
55         printf("Nothing to change!");
56         return 0;
57     }
58
59
60     printf("A = ");
61     printIntArray(k, a);
62
63     return 0;
64 }

```

Файл "safeIO.h"

```

1  #include <stdio.h>
2  #include <stdarg.h>
3  #include <stdlib.h>
4  #include <limits.h>
5  #include <float.h>
6  #include <math.h>
7
8
9  void clearBuffer() {
10     while(getchar() != '\n');
11 }
12
13 int* _enterInts(int count, int min_limit, int max_limit) {
14     int *nums = malloc(sizeof(int)*count);
15     float input;
16     int inputCode;
17
18     for (int i=0; i<count; i++) {
19         inputCode = scanf("%f", &input);
20
21         if (!inputCode || input < min_limit || input > max_limit || ((int)input) != input) {
22             clearBuffer();
23             printf("Wrong input! Enter %d more numbers!\n", count-i);
24             i--;
25         }
26         else
27             nums[i] = (int)input;
28     }
29
30     return nums;
31 }
32
33 void enterIntRanged(int *dir, int minLimit, int maxLimit) {
34     int *nums = _enterInts(1, minLimit, maxLimit);
35     *dir = nums[0];
36     free(nums);
37 }
38
39 void enterInt(int *dir) {
40     int *nums = _enterInts(1, INT_MIN, INT_MAX);
41     *dir = nums[0];
42     free(nums);
43 }

```

```
44
45 void enterInts(int count, ...) {
46     va_list arg;
47     va_start(arg, count);
48
49     int *nums = _enterInts(count, INT_MIN, INT_MAX);
50     for (int i=0; i<count; i++)
51         *va_arg(arg, int*) = nums[i];
52     free(nums);
53     va_end(arg);
54 }
55
56 void enterIntsRanged(int minLimit, int maxLimit, int count, ...) {
57     va_list arg;
58     va_start(arg, count);
59     int *nums = _enterInts(count, minLimit, maxLimit);
60     for (int i=0; i<count; i++)
61         *va_arg(arg, int*) = nums[i];
62     free(nums);
63     va_end(arg);
64 }
65
66 void enterIntArrayRanged(int *ptr, int minLimit, int maxLimit, int len) {
67     int *nums = _enterInts(len, minLimit, maxLimit);
68     for (int i=0; i<len; i++)
69         ptr[i] = nums[i];
70     free(nums);
71 }
72
73 void enterIntArray(int *ptr, int len) {
74     enterIntArrayRanged(ptr, INT_MIN, INT_MAX, len);
75 }
76
77
78
79
80 float *_enterFloats(int count, float min_limit, float max_limit) {
81     float *nums = malloc(sizeof(float)*count);
82     double input;
83     int inputCode;
84
85     for (int i=0; i<count; i++) {
86         inputCode = scanf("%lf", &input);
87
88         if (!inputCode || input < min_limit || input > max_limit) {
89             clearBuffer();
90             printf("Wrong input! Enter %d more numbers:\n", count-i);
91             ;
92             i--;
93         }
94         else
95             nums[i] = (float)input;
96     }
97
98     return nums;
99 }
100
101 void enterFloatRanged(float *ptr, float min_val, float max_val) {
102     float *nums = _enterFloats(1, min_val, max_val);
103     *ptr = nums[0];
104     free(nums);
105 }
106
107 void enterFloat(float *ptr) {
```

```
109     float *nums = _enterFloats(1, FLT_MIN, FLT_MAX);
110     *ptr = nums[0];
111     free(nums);
112 }
113
114 void enterFloatsRanged(float min_val, float max_val, int count, ...) {
115     va_list arg;
116     va_start(arg, count);
117     float *nums = _enterFloats(count, min_val, max_val);
118
119     for (int i=0; i<count; i++) {
120         float *ptr = va_arg(arg, float*);
121         *ptr = nums[i];
122     }
123
124     free(nums);
125     va_end(arg);
126 }
127
128 void enterFloats(int count, ...) {
129     va_list arg;
130     va_start(arg, count);
131     float *nums = _enterFloats(count, FLT_MIN, FLT_MAX);
132
133     for (int i=0; i<count; i++) {
134         float *ptr = va_arg(arg, float*);
135         *ptr = nums[i];
136     }
137
138     free(nums);
139     va_end(arg);
140 }
141
142
143 void printIntArray(int len, int *arr) {
144     printf("{");
145     if (len < 1) {
146         printf("}");
147         return;
148     }
149     for (int i=0; i<len-1; i++)
150         printf("%d, ", arr[i]);
151     printf("%d}", arr[len-1]);
152 }
153
154 void enterString(char *ptr, int len) {
155     clearBuffer();
156
157     char sym;
158     for (int i=0; i<len-1; i++) {
159         scanf("%c", &sym);
160         if (sym == '\n') {
161             ptr[i] = '\0';
162             return;
163         }
164         ptr[i] = sym;
165     }
166     ptr[len-1] = '\0';
167 }
168
169 void printString(char *ptr, int len) {
170     for (int i=0; i<len; i++)
171         printf("%c", ptr[i]);
172 }
173 }
```

Тесты

Задание I

№	Входные данные	Выходные данные
1	$x_1 = 1 \ x_2 = 5 \ x_3 = 10$ $y_1 = 1 \ y_2 = 2 \ y_3 = 3$ $x=1$	$f(1) = 1.000000$
2	$x_1 = 1 \ x_2 = 5 \ x_3 = 10$ $y_1 = 1 \ y_2 = 2 \ y_3 = 3$ $x=2$	$f(2) = 1.250000$

Задание II

№	Входные данные	Выходные данные
1	$n=2, m=2$ $B = [5, 2]$ $[10, 1]$	$H = 2$
2	$n=1, m=2$ $B = [1, 2]$	$H = 1$

Задание III

№	Входные данные	Выходные данные
1	$n=5$ $A=[1, 2, 3, 4, 5]$	$A = \{\}$
2	$n=5$ $A=[1, 1, 2, 2, 3]$	$A = \{1, 1, 2, 2\}$
3	$n=4$ $A=[1, 1, 2, 2]$	Nothing to change!