

# Алгоритмизация и программирование

## Ерохина Елена Альфредовна

### Контрольные точки

#### 1 модуль:

1. 10 лекций (проверочные работы или тесты)
2. 2 лабораторные работы
3. 4 семинара (проверочные работы или тесты)

#### 2 модуль:

1. 12 лекций (проверочные работы)
2. 3 лабораторные работы
3. 7 семинаров (проверочные работы или тесты)
4. Контрольная работа (на семинаре)

# Где найти материалы курса

Материалы курса размещаются в Smart Lms. Учебный офис автоматически регистрирует студентов на курс.

При возникновении вопросов по регистрации обращайтесь в учебный офис к менеджеру образовательной программы.

Оценка за текущий контроль в 1 и 2 модуле учитывает результаты студента следующим образом.

Модуль 1.  $O_{\text{текущая 1}} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}}$

Модуль 2.  $O_{\text{текущая 2}} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}} + O_{\text{контр. работа}}$

Все оценки рассматриваются без округления.

Результирующая оценка учитывает оценки модулей. Промежуточная оценка за 1 и 2 модуль вычисляется по формуле

$O_{\text{промежуточная 1 и 2}} = 0,3 * O_{\text{текущая 1}} + 0,5 * O_{\text{текущая 2}} + 0,2 * O_{\text{экзамен 2 модуль}}$ ,

где  $O_{\text{текущая 1}}$ ,  $O_{\text{текущая 2}}$  – оценки текущего контроля 1, 2 модуля, без округления.

Округление производится один раз, после вычисления промежуточной оценки, по правилам арифметики.

Экзаменационная оценка не является блокирующей. Промежуточная оценка за 1 и 2 модуль не может превышать 10 баллов, в случае превышения ставится промежуточная оценка 10 баллов.

Результирующая оценка за дисциплину вычисляется по формуле

$$O_{\text{результатирующая}} = 0,4 * O_{\text{промежуточная 1 и 2}} + 0,6 * O_{\text{промежуточная 3 и 4}}$$

Округление производится по правилам арифметики.

В диплом выставляется результирующая оценка.

Для вычисления текущей оценки по дисциплине используется следующая таблица (для групп БИВ).

	Работа на семинарском занятии	Работа на лекции	Выполнение лабораторного практикума	Контрольная работа
1 модуль	1	2	7 (3+4)	-
2 модуль	1	1	5(2+2+1)	3
3 модуль	1	1	5(1,5+1,5+2)	3
4 модуль	1	1	5(2,3+1,7+1)	3

Для вычисления текущей оценки по дисциплине используется следующая таблица (для групп БИТ).

	Работа на семинарском занятии	Работа на лекции	Выполнение лабораторного практикума	Контрольная работа
1 модуль	1	2	7 (3+4)	-
2 модуль	1	1	5(2+2+1)	3

В скобках указано распределение баллов по лабораторным работам.

- Ни один из элементов текущего контроля не является блокирующим.
- На некоторых семинарах и лекциях проводится тест или проверочная работа. Каждый вид работы оценивается от 1 до 4 баллов. В итоговую оценку эти баллы входят с коэффициентом, получаемым делением числа занятий, на которых проводилось оценивание, на общее количество занятий.
- При пропуске лекции или семинарского занятия по любой причине студент не может решить дополнительное задание для компенсации баллов, которые он мог бы получить на этом занятии.
- Кроме того, преподаватель может оценивать дополнительными баллами ответ студента у доски (максимум 2 балла) и активное участие в решении задач семинаров (например, выявление и исправление неточностей и ошибок в алгоритмах и при кодировании программ, внесение усовершенствований в алгоритм и т.п.) (максимум по 0.2 балла за каждый ответ).

- Для каждой лабораторной работы устанавливается срок защиты отчета (в 1 модуле на 2 и 4 занятия, считая пары отдельно для каждой подгруппы). При своевременной защите работа оценивается полученным баллом, при опоздании на 1 неделю балл снижается на 40%, при опоздании на 2 недели балл снижается на 60% от полученной оценки. При опоздании более чем на 2 недели работа не оценивается.
- В случае пропуска занятий по уважительной причине (обязательно предоставление справки) срок сдачи лабораторной работы может быть перенесен на соответствующее количество рабочих дней.
- В случае пропуска занятий по уважительной причине (обязательно предоставление справки) предоставляется дополнительное время для написания контрольной работы (единственная дата переписывания заранее сообщается через старост).
- Переписывание контрольной работы с целью повышения полученной оценки не допускается.



# Требования к оформлению отчета

Выполнение каждой работы лабораторного практикума завершается написанием отчета, включающего следующие разделы:

- титульный лист
- содержание;
- задание;
- постановка задачи - 0,5;
- метод решения задачи - 1;
- внешняя спецификация - 0,5;
- описание алгоритма на псевдокоде - 1,5;
- листинг программы - 0,5 +1 программа работает;
- распечатка тестов к программе и результатов – 1;
- вопросы по отчету – 2;
- дополнительное задание – 2.

Через тире указан вес каждого раздела в оценке за выполнение лабораторной работы.

# Требования к оформлению отчета

- Для оформления отчета используется формат бумаги А4.
- Печать отчета предусматривает распечатку титульного листа и страницы с заданием. Остальная часть отчета предъявляется для защиты в электронном виде.
- Каждый раздел отчета начинается с нового листа.
- Правки в итоговый отчет нужно внести после исправления всех замечаний.
- Итоговый отчет отправляется на почту преподавателю (или учебному ассистенту), принимавшему отчет.
- Итоговая оценка выставляется на титульном листе отчёта. Титульный лист сдается преподавателю.
- Студенту необходимо сделать фото титульного листа с оценкой и сохранять его до конца модуля.

**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)

**Курс: Алгоритмизация и программирование**

Раздел	Мак 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	

**ОТЧЕТ  
по лабораторной работе №\_\_**

Студент: \_\_\_\_\_ **ФИО** \_\_\_\_\_

Группа: \_\_\_\_\_ **номер группы** \_\_\_\_\_

Вариант: **№ (номера заданий)** \_\_\_\_\_

Руководитель: \_\_\_\_\_

Оценка: \_\_\_\_\_

Дата сдачи \_\_\_\_\_

**МОСКВА 2022**

# Пример оформления отчета по лабораторной работе 1

## Задание.

1. Даны  $n$ ,  $x$ ,  $h$ ,  $a$ . Вычислить массив  $R[1:n]$  в соответствии с формулами:  $R[i]=2,5\sin(ax+i^2h)$ ,  $i=\overline{1, n}$
2. Из вычисленного массива  $R$  удалить все отрицательные элементы, расположенные между первым минимальным и последним положительным элементами.
3. В полученном массиве  $R[1:k]$ , где  $k$  – число элементов, оставшихся после удаления, подсчитать среднее арифметическое элементов, расположенных до первого отрицательного элемента.

# Постановка задачи

Дано:

1.  $n$ -цел.,  $x$ ,  $h$ ,  $a$ -вещ.
2. Нет входных данных
3. Нет входных данных

Результат:

1.  $R[1:n]$ -вещ.
2.  $R[1:k]$ -вещ. или сообщение <<Нет положительного элемента>> или сообщение <<Первый минимальный и последний положительный расположены рядом или совпадают>>
3.  $sr$ -вещ. или сообщение <<Нет среднего значения>>

При:  $n \in \mathbb{N}, n \leq lmax$ .

Связь:

1. См. формулу в условии

2.  $\exists n1: \forall i = \overline{1, n}$

$$R[n1] \leq R[i]$$

$$\min = R[n1]$$

$$\overline{\exists t: t = \overline{1, n1 - 1}; R[t] = \min}$$

$$\exists np: np = \overline{1, n}; R[np] > 0, \exists q: q = \overline{np + 1, n}; R[q] > 0$$

$$c = \min(n1, np); b = \max(n1, np)$$

$$\forall i = \overline{c + 1, b - 1}: R[i] \geq 0 \quad \exists t \in [c + 1, l]: R[t] = R[i]$$

$$\forall i = \overline{b, n} \exists p \in [l + 1, k]: R[p] = R[i]$$

3.  $\exists n1: n1 = \overline{1, k}; R[n1] < 0, \exists t: t = \overline{1, n1 - 1}; R[t] < 0$

$$sr = \sum_{i=1}^{n1-1} R[i]/(n1 - 1)$$

# Метод решения задачи

- $\left\{ \begin{array}{l} \text{для } i = \overline{1, n} \\ r[i] = 2,5\sin(ax + i^2h) \end{array} \right.$
- $np=0$   
 $n1=1$   
 $k=n$   
 $\left\{ \begin{array}{l} \text{для } i = \overline{1, n} \\ np = i, \text{ если } r[i] > 0; n1 = i, \text{ если } r[i] < r[n1] \end{array} \right.$   
 $c=n1; b=np, \text{ если } n1 < np$   
 $c=np; b=n1, \text{ если } np \leq n1$   
 $k=c$   
 $\left\{ \begin{array}{l} \text{для } i = \overline{c+1, b-1} \\ k = k+1; r[k] = r[i], \text{ если } r[i] \geq 0 \end{array} \right.$   
 $\left\{ \begin{array}{l} \text{для } i = \overline{b, n} \\ k = k+1; r[k] = r[i] \end{array} \right.$
- $n1=0$   
 $\left\{ \begin{array}{l} \text{для } i = \overline{k, 1} \\ n1 = i, \text{ если } r[i] < 0 \end{array} \right.$   
 $sr=0$   
 $\left\{ \begin{array}{l} \text{для } i = \overline{1, n1-1} \\ sr = sr + r[i] \end{array} \right.$   
 $sr = sr / (n1 - 1)$

# Внешняя спецификация

Лабораторная работа №1

Задание 1

Введите длину массива R от 1 до <<Imax>>:

$\{\langle n \rangle\}^*$  до  $n > 0$  и  $n \leq \text{Imax}$

Введите x, h, a:

<x> <h> <a>

Массив R из <<n>> элементов

<<R[1]>> <<R[2]>>...<<R[n]>>

## Задание 2

При  $n_1 = 0$

{ Нет положительного элемента

Иначе

{ при  $|n_1 - n_2| < 2$

{ Первый минимальный и последний положительный  
расположены рядом или совпадают

{ иначе

{ Массив  $r$  состоит из  $\ll k \gg$  элементов  
 $\ll r[1] \gg \ll r[2] \gg \dots \ll r[k] \gg$

## Задание 3

при  $n_1 - 1 \leq 0$

{ Нет среднего значения

иначе

{  $sr = \ll sr \gg$



# Описание алгоритма на псевдокоде

Алг «Лабораторная работа №1»

нач

**{задание 1}**

{ввод исходных данных для задания 1}

вывод(« Лабораторная работа №1.Задание 1»)

вывод(«Введите длину массива R от 1 до », lmax)

цикл

ввод(n)

до  $n > 0$  и  $n \leq lmax$

кц

вывод(«Введите x, h, a:»)

ввод(x, h, a)

цикл от  $i := 1$  до n

$r[i] := 2,5 \sin(ax + hi^2)$

кц

вывод(«Массив R из », n, « элементов: »)    вывод( $r[1:n]$ )

## {Задание 2}

вывод(«Задание 2»)

$k:=n$  {число элементов, оставшихся после удаления}

$n1:=1$  { номер первого минимального элемента}

$nr:=0$  { номер последнего положительного элемента}

цикл от  $i:=1$  до  $n$

если  $r[i]>0$  то  $nr:=i$

все

если  $r[i]<r[n1]$  то  $n1:=i$

все

кц

{анализ существования результата и вывод результата задания 2}

если  $nr=0$  то

вывод(«Нет положительного элемента»)

иначе

если  $|nr-n1|<2$  то

вывод(«Первый минимальный и последний положительный  
расположены рядом или совпадают»)

иначе

{с и b – начало и конец зоны удаления}

если  $n1 < np$  то

$c := n1$ ;  $b := np$

иначе

$c := np$ ;  $b := n1$

все

$k := c$

цикл от  $i := c + 1$  до  $b - 1$

если  $r[i] \geq 0$  то

$k := k + 1$

$r[k] := r[i]$

все

кц

цикл от  $i := b$  до  $n$

$k := k + 1$

$r[k] := r[i]$

кц

    Вывод («Массив r из », k, « элементов»)

    Вывод ( $r[1:k]$ )

все

все

### {Задание3}

вывод(«Задание 3»)

$n1:=0$  {номер первого отрицательного элемента}

цикл от  $i:=k$  до 1 шаг -1

если  $r[i]<0$  то

$n1:=i$

все

кц

если  $n1-1 \leq 0$  то

вывод(«Нет среднего значения»)

иначе

$sr:=0$

цикл от  $i:=1$  до  $n1-1$

$sr:=sr+r[i]$

кц

$sr:=sr/(n1-1)$

Вывод( $sr$ )

все

кон

# Листинг программы

```
#include <stdio.h>
#include <math.h>
#define lmax 200
int main()
{

int n, c, b, n1, np, i, k;
float x, h, a, sr, r[lmax];
    printf («Лабораторная работа №1\n»);
    printf («Задание №1\n»);
```

```
//Задание 1
do
    {printf («Введите длину массива R от 1 до
%d:", lmax);
    scanf ("%d", &n);
}
while (n <= 0 || n > lmax);
printf («Введите x, h, a:");
scanf ("%f", &x);
scanf ("%f", &h);
scanf ("%f", &a);
for( i = 1; i<=n; i++)
    r[i] = 2.5 * sin(a * x + h * i * i);
printf («Массив R из %d элементов:\n");
for( i = 1; i<=n; i++)
    printf ("%8.3f ", r[i]);
printf ("\n");
```

```
//задание 2
k = n;
printf («Задание №2\n»);
for(n1 = 1,np = 0,i = 1;i<=n;i++)
{
    if (r[i] > 0 )
        np = i;
    if (r[i] < r[n1])
        n1 = i;
}
if (np == 0 )
    printf(" Нет положительного элемента \n");
else
    if (abs(np - n1) < 2 )
        printf(" Первый минимальный и последний
положительный расположены рядом или совпадают
\n");
```

```
else
{
    if (n1 < np )
    {
        c = n1;
        b = np;
    }
    else
    {
        c = np;
        b = n1;
    };
    k = c;
    for (i = c + 1; i <= b - 1; i++)
        if (r[i] >= 0 )
        {
            k = k + 1;
            r[k] = r[i];
        }
}
```



```
for (i = b; i<= n; i++)
{
    k = k + 1;
    r[k] = r[i];
}
    printf («Массив R из %d
элементов:\n", k);
    for( i = 1; i<=k; i++)
        printf ("%8.3f ", r[i]);
    printf ("\n");
};
```

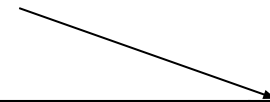
```
//Задание 3
    printf(" Задание №3\n");
for(n1 = 0,i = k;i>=1; i--)
    if (r[i] < 0 )
        n1 = i;
if (n1 - 1 <= 0 )
    printf(" Нет среднего значения ");
else
{ sr = 0;
  for( i = 1;i<=n1-1; i++)
      sr = sr + r[i];
  sr = sr / (n1 - 1);
  printf("sr=%8.3f\n", sr);
}
}
```

# Распечатка тестов к программе и результатов

№	Исходные данные	Результаты
1	n=10 x=2; h=4; a=5	R={2.477; 1.906; 2.254; -2.463; -0.111; -0.155; -2.437; 2.150; 2.092; 2.499} R={2.477; 1.906; 2.254; -2.463; 2.150; 2.092; 2.499} Sr=2.212
2	n=1 x=5; h=8; a=7	R={-2.079} Нет положительного элемента Нет среднего значения
3	n=5 x=5; h=2; a=4	R={-0,022; 0.677; 0.741; 2.467; 1.935} R={-0,022; 0.677; 0.741; 2.467; 1.935} Нет среднего значения

Для выбора набора заданий используйте формулы:

Пример



№ варианта	X – номер варианта	X=13
Задание 1	$(\text{Остаток от деления } x \text{ на } 7) + 1$	7
Задание 2	$(\text{Остаток от деления } x \text{ на } 9) + 1$	5
Задание 3	$(\text{Остаток от деления } x \text{ на } 10) + 1$	4

7, 9, 10 – количество вариантов заданий в л.р. 1.

Номера вариантов указаны в журнале на страницах групп. Также список номеров вариантов есть в Smart Ims.

Методические указания по лабораторному практикуму размещены в Smart Ims