



Федеральное агентство по рыболовству  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Калининградский государственный технический университет»  
(ФГБОУ ВО «КГТУ»)

УТВЕРЖДАЮ  
Начальник УРОПСИ

Фонд оценочных средств  
(приложение к рабочей программе модуля)  
**«ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ И АЛГОРИТМИЗАЦИИ»**

основной профессиональной образовательной программы магистратуры  
по направлению подготовки

**09.04.01 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА**

ИНСТИТУТ  
РАЗРАБОТЧИК

цифровых технологий  
кафедра прикладной математики и информационных технологий

## 1 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Таблица 1 – Планируемые результаты обучения по дисциплине, соотнесенные с установленными индикаторами достижения компетенций

Код и наименование компетенции	Индикаторы достижения компетенции	Дисциплина	Результаты обучения (владения, умения и знания), соотнесенные с компетенциями/индикаторами достижения компетенции
<p>ОПК-2: Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p>	<p>ОПК-2.2: Разрабатывает оригинальные алгоритмы и программные средства на языках высокого уровня в соответствии с поставленной задачей</p>	<p>Практикум по программированию и алгоритмизации</p>	<p><u>Знать:</u></p> <ul style="list-style-type: none"> <li>- современный опыт использования технологий больших данных в части разработки новых методов, моделей, алгоритмов, технологий и инструментальных средств;</li> <li>- синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования;</li> <li>- технологии программирования;</li> <li>- особенности выбранной среды программирования;</li> </ul> <p><u>Уметь:</u></p> <ul style="list-style-type: none"> <li>- применять выбранные языки программирования для написания программного кода;</li> <li>- использовать выбранную среду программирования и средства системы управления базами данных;</li> <li>- использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода;</li> <li>- осуществлять коммуникации с заинтересованными сторонами;</li> </ul> <p><u>Владеть:</u></p> <ul style="list-style-type: none"> <li>- навыками создания программного кода в соответствии с техническим заданием (спецификацией);</li> <li>- навыками оптимизация программного кода.</li> </ul>

## 2 ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ И КРИТЕРИИ ОЦЕНИВАНИЯ

2.1 К оценочным средствам текущего контроля успеваемости относятся:

- тестовые задания открытого и закрытого типов.

2.2 Промежуточная аттестация по дисциплине проводится в форме зачета, который выставляется по результатам прохождения всех видов текущего контроля успеваемости. При

необходимости тестовые задания закрытого и открытого типов могут быть использованы для проведения промежуточной аттестации.

### 2.3 Критерии оценки результатов освоения дисциплины

Универсальная система оценивания результатов обучения включает в себя системы оценок: 1) «отлично», «хорошо», «удовлетворительно», «неудовлетворительно»; 2) «зачтено», «не зачтено»; 3) 100 – балльную/процентную систему и правило перевода оценок в пятибалльную систему (табл. 2).

Таблица 2 – Система оценок и критерии выставления оценки

Система оценок	2	3	4	5
	0-40%	41-60%	61-80 %	81-100 %
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»
	«не зачтено»	«зачтено»		
<b>Критерий</b>				
<b>1 Системность и полнота знаний в отношении изучаемых объектов</b>	Обладает частичными и разрозненными знаниями, которые не может научно-корректно связывать между собой (только некоторые из которых может связывать между собой)	Обладает минимальным набором знаний, необходимым для системного взгляда на изучаемый объект	Обладает набором знаний, достаточным для системного взгляда на изучаемый объект	Обладает полнотой знаний и системным взглядом на изучаемый объект
<b>2 Работа с информацией</b>	Не в состоянии находить необходимую информацию, либо в состоянии находить отдельные фрагменты информации в рамках поставленной задачи	Может найти необходимую информацию в рамках поставленной задачи	Может найти, интерпретировать и систематизировать необходимую информацию в рамках поставленной задачи	Может найти, систематизировать и систематизировать необходимую информацию, а также выявить новые, дополнительные источники информации в рамках поставленной задачи
<b>3. Научное осмысление изучаемого явления, процесса, объекта</b>	Не может делать научно корректных выводов из имеющихся у него сведений, в состоянии проанализировать только некоторые	В состоянии осуществлять научно корректный анализ предоставленной информации	В состоянии осуществлять систематический и научно корректный анализ предоставленной информации,	В состоянии осуществлять систематический и научно-корректный анализ предоставленной информации,

Система оценок	2	3	4	5
	0-40%	41-60%	61-80 %	81-100 %
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»
Критерий	«не зачтено»	«зачтено»		
	из имеющихся у него сведений		вовлекает в исследование новые релевантные задаче данные	вовлекает в исследование новые релевантные поставленной задаче данные, предлагает новые ракурсы поставленной задачи
<b>4. Освоение стандартных алгоритмов решения профессиональных задач</b>	В состоянии решать только фрагменты поставленной задачи в соответствии с заданным алгоритмом, не освоил предложенный алгоритм, допускает ошибки	В состоянии решать поставленные задачи в соответствии с заданным алгоритмом	В состоянии решать поставленные задачи в соответствии с заданным алгоритмом, понимает основы предложенного алгоритма	Не только владеет алгоритмом и понимает его основы, но и предлагает новые решения в рамках поставленной задачи

### 3 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

ОПК-2: Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач.

Индикатор ОПК-2.2: Разрабатывает оригинальные алгоритмы и программные средства на языках высокого уровня в соответствии с поставленной задачей.

#### Тестовые задания открытого типа:

1. Для решения задачи прогнозирования в сезонных временных рядах предназначена модель \_\_\_\_\_

*Регистр важен.*

**Ответ: SARIMA**

2. Структура, компонентами которой являются взаимосвязанные данные различных типов и использующие эти данные процедуры и функции – это \_\_\_\_\_

**Ответ: объект**

3. Процесс модификации существующей программы для ЭВМ, обусловленный необходимостью устранения выявленных в ней ошибок и (или) изменения ее функциональных возможностей – это \_\_\_\_\_

**Ответ: сопровождение**

4. \_\_\_\_\_ анализ имеет основной целью выявление взаимосвязей между переменными

**Ответ: факторный**

5. Парная линейная регрессия применима при условии, если между зависимой и независимой переменными имеется \_\_\_\_\_

**Ответ: корреляция**

6. Линейная регрессия, в которой в качестве входных используются лаговые переменные – это \_\_\_\_\_

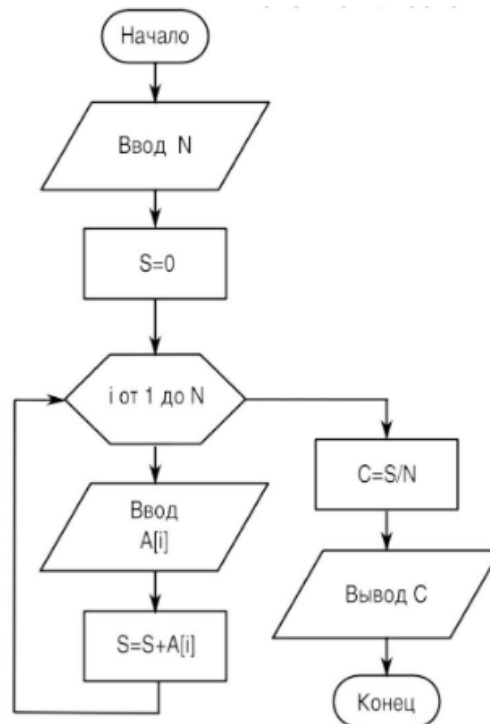
*Укажите вид регрессии в именительном падеже, регистр не важен*

**Ответ: авторегрессия**

7. Не имеет тренда \_\_\_\_\_ временной ряд.

**Ответ: стационарный**

8. Задана блок-схема алгоритма.



При значениях  $N=10$ ,  $A=\{0,1,2,3,4,5,6,7,8,9\}$  переменная  $C$  будет равна \_\_\_\_\_.  
*Введите число (разделитель разрядов – запятая)*

**Ответ:4,5**

9. В программный код (язык C):

`char a[10],b[10];`

`strcpy(a,b);`

для работы функции необходимо добавить директиву \_\_\_\_\_

**Ответ: #include <string.h>**

10. В программный код (язык C):

```
double d = sin(1.0);
```

для работы функции необходимо добавить директиву \_\_\_\_\_

**Ответ: #include <math.h>**

**Тестовые задания открытого типа (на дополнение):**

1. В приведенном ниже фрагменте кода (язык Python)

```
from statsmodels.tsa.arima.model import ARIMA
```

```
model = ARIMA(series, order=(5,1,0))
```

параметр авторегрессии в модели ARIMA равен \_\_\_\_

**Ответ: 5**

2. В результате выполнения программного кода (язык C):

```
#include <stdio.h>
```

```
#define n 5
```

```
void fun(int *mas, int k);
```

```
int sm_1(int *mas, int k);
```

```
int main( void )
```

```
{
```

```
    int arr[n] = {3,4,1,2};
```

```
    int i, s;
```

```
    fun(arr, n);
```

```
    s = sm_1(arr, n);
```

```
    return 0;
```

```
}
```

```
void fun(int *mas, int k)
```

```
{
```

```
    int i, s = 0;
```

```
    for (i=0; i<k; i++) ++mas[i];
```

```
}
```

```
int sm_1(int *mas, int k)
```

```
{
```

```
    int i, s = 0;
```

```
    for (i=0; i<k; i++) s += mas[i];
```

```
    return s;
```

```
}
```

значение переменной s будет равно \_\_\_\_

**Ответ: 15**

3. В файле primer.h, расположенном в корневом каталоге диска E:, есть описания переменных и функций:

```
extern int x, y;
extern char ch;

void func22(void)
{
    y = 100;
}

void func23(void)
{
    x = y/10;
    ch = 'R';
}
```

Использовать эти описания в другом файле возможно при наличии директивы препроцессору \_\_\_\_\_

*Введите директиву (регистр важен)*

**Ответ: #include "E:\primer.h"**

### **Тестовые задания открытого типа (с развернутым ответом):**

1. Перечислите основные области (2-8 областей) применения метода частиц в ячейках

**Правильный ответ:**

- a. Физика плазмы**
- b. Гидродинамика**
- c. газовая динамика**
- d. Астрофизика**
- e. Физика полупроводников**
- f. Геофизика**
- g. Физика элементарных частиц**
- h. Квантовая химия**

*Критерии оценивания:*

«зачтено» - если приведены от 2 до 8 перечисленных областей науки

«незачтено» - приведена 1 область или ни одной.

2. Приведите алгоритм вычисления производительности процессора, определенную на примере программы, реализующей метод частиц в ячейках?

**Правильный ответ: Умножить количество операций, выполняемых с одной модельной частицей на количество модельных частиц и поделить на время расчета движения модельных частиц**

*Критерии оценивания:*

«зачтено» - если упомянуто время расчета движения модельных частиц;

«зачтено» - если упомянуто количество модельных частиц;

«зачтено» - если к тому же упоминается и количество операций на одну модельную частицу;

«незачтено» - если не упоминается время расчета движения модельных частиц.

3. Перечислите основные причины (1-3), делающие метод частиц в ячейках популярным методом решения задач физики плазмы

**Правильный ответ:**

**1. Относительная простота**

**2. Легкость реализации на суперЭВМ**

**3. Возможность обойтись без нефизических упрощений**

*Критерии оценивания:*

«зачтено» - если приведены 1, 2 или 3 из перечисленных причин

«незачтено» - если не упоминается ни одной из причин.

**Тестовые задания закрытого типа (с одним вариантом ответа):**

1. Модульное программирование – это такая технология программирования, когда:

а. программа разбивается на связанные между собой процедуры и функции

б. исходный алгоритм разделяется на основной модуль и подпрограммы, вызываемые из основного модуля

**в. отдельные процедуры и функции объединяются в групповые разделы (модули)**

г. алгоритм исходной задачи разбивается на более простые подзадачи, компилированные отдельно

2. При объектно-ориентированном подходе простое и множественное наследование является примером иерархии

**а. классов**

б. объектов

в. методов

г. модулей

3. API-интерфейс (Application Program Interface) строится на основе диаграммы:

**а. классов**

б. объектов

в. случаев (интерфейсов)

г. модулей

4. Язык C: функция f1() выводит на экран переданное ей значение.

```
#include <stdio.h>
```

```
struct struct_type {
```

```
    int a, b;
```

```
    char ch;
```

```
};
```

```
int main(void)
```



```
{
    struct struct_type arg;
    arg.a = 1000;
    f1(&arg);
    return 0;
}
```

Для корректного выполнения приведенной программы правильным описанием функции f1 будет:

- а. void f1(struct struct\_type \*parm) { printf("%d", parm.a); }
- б. void f1(struct struct\_type \*parm) { printf("%d", parm->a); }**
- в. void f1(struct struct\_type parm) { printf("%d", parm->a); }
- г. void f1(struct struct\_type \*parm) { printf("%d", &parm.a); }

5. На примере программы, реализующей метод частиц в ячейках, вычислительная производительность процессора определяется:

- а. умножением количества модельных частиц на количество операций с плавающей точкой, необходимое для расчета движения одной модельной частицы, и делением на время, затраченное на расчет движения модельных частиц**
- б. делением количества модельных частиц на время, затраченное на расчет движения модельных частиц
- в. номинальной производительностью процессора
- г. делением количества операций с плавающей точкой, необходимое для расчета движения одной модельной частицы, на время, затраченное на расчет движения модельных частиц

6. Построение количественной зависимости производительности суперЭВМ от наиболее важных факторов целесообразно выполнять с помощью:

- а. множественной линейной регрессии
- б. регрессионной нейронной сети
- в. факторного анализа**
- г. метода главных компонент

7. С помощью модели ARIMA **НЕ** могут быть решены задачи регрессии:

- а. с резко нелинейной зависимостью**
- б. с большим количеством независимых переменных
- в. во временных рядах с сильной автокорреляцией
- г. в быстро осциллирующих временных рядах

8. В приведенном фрагменте кода (язык C)

```
...
int test();
int a = 1;
int main (void)
{
    int b, s;
```

```
extern int x;
static int c;
s = test();
return 0;
}
```

```
int test()
```

```
{
    int t = a;
    return t;
}
```

```
int x = 4;
```

...

память выделяется при входе в блок и освобождается при выходе из него для локальной переменной:

а. а

**б. b**

в. x

г. c

### Тестовые задания закрытого типа (на последовательность и соответствие):

1. Установите соответствие между заголовочными файлами языка C и задачами, которые могут решаться с помощью функций, описанных в этих файлах

- |             |  |
|-------------|--|
| 1. math.h   | а. Работа с динамической памятью (в том числе) |
| 2. stdio.h  | б. математические операции                     |
| 3. string.h | с. ввод-вывод                                  |
| 4. stdlib.h | д. работа со строками                          |

**Правильный ответ: 1-б, 2-с, 3-д, 4-а**

2. Установите соответствие между методами анализа данных и задачами, для решения которых они предназначены

- |                                     |  |
|-------------------------------------|--|
| 1. Множественная линейная регрессия | а. Уточнение количества независимых переменных   |
| 2. Регрессионная нейронная сеть     | б. Предсказание значений некоторой величины в зависимости от набора независимых переменных |

3. Факторный анализ
- с. Предсказание значений некоторой величины в зависимости от набора независимых переменных в предположении о линейном характере зависимости

**Правильный ответ: 1-с, 2-б, 3-а**

3. Установите соответствие между этапами метода частиц в ячейках и используемыми на данных этапах численными методами:

- |                                     |  |
|-------------------------------------|--|
| 1. инициализация модельных частиц   | a. Схема с перешагиванием                            |
| 2. расчет электрического поля       | b. Генератор случайных чисел                         |
| 3. расчет движения модельных частиц | d. линейная интерполяция                             |
| 4. расчет плотности заряда          | c. Решение системы линейных алгебраических уравнений |

**Правильный ответ: 1-б, 2-с, 3-а, 4-d**

4. Укажите правильную последовательность действий при решении задачи методом частиц в ячейках

- a. Расчет плотности заряда
- b. Расчет движения модельных частиц
- c. Инициализация модельных частиц
- d. Расчет электрического поля

**Правильный ответ: c, a, d, b**

5. Расположите методы решение системы линейных алгебраических уравнений по скорости работы (по номером 1 поставьте наиболее быстрый метод)

- a. Метод Гаусса-Зайделя
- b. Метод Крамера
- c. Метод сопряженных градиентов

**Правильный ответ: 1-с, 2-а, 3-б**

6. Расположите характеристики суперЭВМ по степени влияния на производительность суперЭВМ в целом

- e. Производительность системы памяти
- f. Производительность процессора
- g. Производительность сетевой подсистемы

h. Производительность жесткого диска

**Правильный ответ: 1-a, 2-c, 3-b, 4-d**

#### **4 ТИПОВЫЕ ЗАДАНИЯ НА КОНТРОЛЬНУЮ РАБОТУ, КУРСОВУЮ РАБОТУ/КУРСОВОЙ ПРОЕКТ**

Данный вид контроля по дисциплине не предусмотрен учебным планом.

## **5 СВЕДЕНИЯ О ФОНДЕ ОЦЕНОЧНЫХ СРЕДСТВ И ЕГО СОГЛАСОВАНИИ**

Фонд оценочных средств для аттестации по дисциплине «Практикум по программированию и алгоритмизации» представляет собой компонент основной профессиональной образовательной программы магистратуры по направлению подготовки 09.04.01 Информатика и вычислительная техника.

Фонд оценочных средств рассмотрен и одобрен на заседании методической комиссии института цифровых технологий (протокол № 2 от 26.04.2022 г.).

Фонд оценочных средств актуализирован. Изменения, дополнения рассмотрены и одобрены на заседании методической комиссии института цифровых технологий (протокол № 3 от 24.03.2023 г.).

Директор института



А.Б. Тристанов