

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ  
Директор института прикладной  
математики и компьютерных наук  
  
А. В. Замятин  
« 16 » июня 2024 г.

Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине  
(Оценочные средства по дисциплине)

**Информатика**

по направлению подготовки / специальности

**10.05.01 Компьютерная безопасность**

Направленность (профиль) подготовки / специализация:  
**Анализ безопасности компьютерных систем**

ОМ составил(и):

канд. физ.-мат. наук, доцент  
доцент кафедры компьютерной безопасности



С.И. Самохина

Рецензент:

канд. физ.-мат., доцент  
доцент кафедры компьютерной безопасности



Н.А. Вихорь

Оценочные средства одобрены на заседании учебно-методической комиссии  
института прикладной математики и компьютерных наук (УМК ИПМКН)

Протокол от 08 июня 2023 г. № 02

Председатель УМК ИПМКН,  
д-р техн. наук, профессор



С.П. Сущенко

**Оценочные средства (ОС)** являются элементом оценивания сформированности компетенций у обучающихся в целом или на определенном этапе ее формирования.

ОС разрабатываются в соответствии с рабочей программой (РП).

### 1. Компетенции и результаты обучения, формируемые в результате освоения дисциплины

Компетенция	Индикатор компетенции	Код и наименование результатов обучения (планируемые результаты обучения, характеризующие этапы формирования компетенций)	Критерии оценивания результатов обучения			
			Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
ОПК-7. Способен создавать программы на языках высокого и низкого уровня, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ИОПК-7.1 Осуществляет построение алгоритма, проведение его анализа и реализации в современных программных комплексах ИОПК-7.2 Понимает общие принципы построения и использования языков программирования высокого уровня и низкого уровня ИОПК-7.3 Демонстрирует навыки создания программ с применением методов и инструментальных средств программирования для решения различных профессиональных, исследовательских и прикладных задач ИОПК-7.4 Осуществляет обоснованный выбор	ОР-7.1.1 Способен разработать алгоритм для решения стандартных задач программирования и реализовать его в виде программы.  ОР-7.2.1 Знает основы низкоуровневого языка программирования Ассемблер. ОР-7.2.2 Знает основы высокоуровневого языка программирования С.  ОР-7.3.1 может создать простую программу на низкоуровневом языке Ассемблер  ОР-7.3.2 может создать простую программу на высокоуровневом языке Ассемблер	Свободно владеет материалом, отвечает на все вопросы	Достаточно свободно владеет материалом, при ответе на вопросы делает небольшие ошибки	Владеет материалом не системно, ошибается при ответе на вопросы	Материалом не владеет, на вопросы не отвечает

	инструментария программирования и способов организации программ	ОР- 7.4.1 Знает устройство современного персонального компьютера. ОР-7.4.2 Знает современный инструментарий программирования и может его использовать.				
--	---	--	--	--	--	--

## 2. Этапы формирования компетенций и виды оценочных средств

№	Этапы формирования компетенций (разделы дисциплины)	Код и наименование результатов обучения	Вид оценочного средства (тесты, задания, кейсы, вопросы и др.)
1.	Введение в информатику	ОР-7.4.1, ОР-7.4.2	Контрольная работа, зачёт
2.	Основы алгоритмизации	ОР-7.1.1	Зачёт
3.	Основы программирования	ОР-7.2.1, ОР-7.2.2, ОР-7.3.1, ОР-7.3.2, ОР-7.1.1	Лабораторные работы, контрольная работа, зачёт
4.	Алгоритмические системы и основные методы трансляции	ОР-7.1.1	Контрольная работа, экзамен
5.	Язык программирования С	ОР-7.2.2, ОР-7.3.2, ОР-7.1.1	Лабораторные работы, контрольная работа, зачёт

## 3. Типовые контрольные задания или иные материалы, необходимые для оценки образовательных результатов обучения

3.1. Типовые задания для проведения текущего контроля успеваемости по дисциплине

Язык низкого уровня Ассемблер:

### Задание 1. Ввод и вывод данных

Ввести три переменные и найти их сумму.

### Задание 2. Арифметические операции

Дано трехзначное число, нужно получить количество десятков в нём.

### Задание 3. Условные переходы

Ввести два числа (до 10) и при необходимости поменять местами их значения так, чтобы в  $a$  было большее число, а в  $b$  меньшее.

### Задание 4. Условные переходы

Среди четырех чисел есть одно, отличное от трёх других. Вывести, какое по счёту число отлично от других. Например, дано 4 4 5 4, выводим 3.

### Задание 5. Организация цикла

Ввести число  $n$  и вычислить  $n!$  Вывести последнюю цифру факториала. При вычислении следить за переполнением. В случае переполнения прервать вычисления и вывести ошибку.

### Задание 6. Вычисление степени

Посчитать значение  $a$  в степени  $k$ , не пользуясь операцией умножения. Предусмотреть все возможные случаи, в том числе частные случаи - 0 и 1. Следить за переполнением, если оно происходит, должен быть выход по ошибке.

Язык высокого уровня С:

### Задание 1. Работа с целочисленной арифметикой.

Дано трехзначное число. Вывести вначале его последнюю цифру (единицы), а затем — его среднюю цифру (десятки).

### Задание 2. Условный оператор

Верно ли высказывание, что только одно из чисел  $x$ ,  $y$ ,  $z$  положительно.

### Задание 3. Оператор цикла

Определить, является ли число  $n$  простым.

**Задание 4.** Вложенные циклы

Дано натуральное число  $Z$ . Если его можно представить в виде суммы двух натуральных чисел в какой-либо степени, то найти пару  $(X, Y)$  таких натуральных чисел, что  $Z = X^n + Y^k$ . Например,  $12 = 2^3 + 4^1$ .

**Задание 5.** Массивы

Из упорядоченных массивов  $A$  длины  $n$  и  $B$  длины  $m$  сформировать упорядоченный массив  $C$  длины  $n+m$ .

**Задание 6.** Массивы

На плоскости заданы  $n$  точек с координатами  $(x[i], y[i])$  и окружность радиуса  $r$ . Определить количество точек, лежащих вне круга и внутри круга.

**Задание 7.** Матрица

Дана прямоугольная матрица. Переставляя ее строки и столбцы переместить наибольший элемент матрицы в верхний левый угол.

**Задание 8.** Булев вектор

Задано множество  $M$ . Представить булевым вектором  $V$  его подмножество  $A$  четных чисел. Например,  $M = \{1, 3, 2, 4, 5, 2, 12\}$ ,  $V = 0011011$ .

**Задание 9.** Динамический массив

Найти сумму положительных элементов заданного массива  $X$ . Решить задачу, используя указатели.

**Задание 9.** Динамический двумерный массив

Дан массив из  $n$  строк. Удалить все строки, содержащие наибольшее число цифр. Массив описать как указатель на указатель.

**Задание 10.** Поиск подстроки в строке

Реализовать алгоритм линейного поиска подстроки в строке. Создать функцию, реализующую данный алгоритм.

**Задание 11.** Ханойские башни

Реализовать алгоритм при произвольном  $n$ .

**Задание 12.** Алгоритмы Маркова

Составить правила и реализовать алгоритм. Инвертировать булев вектор.

**Задание 13.** Файловая сортировка

Реализуем алгоритм прямого слияния на трёх файлах

**Задание 14.** Структуры. Списки

Дан указатель  $P1$  на вершину непустого стека. Извлечь из стека первый (верхний) элемент и вывести его значение  $D$ , а также адрес  $P2$  новой вершины стека. Если после извлечения элемента стек окажется пустым, то положить  $P2 = nil$ . После извлечения элемента из стека освободить память, занимаемую этим элементом

**Задание 15.** Вычисление  $AB$

Арифметическое выражение задано в виде строки. Найти его значение.

3.2. Типовые задания для проведения промежуточной аттестации по дисциплине

Вопросы на зачёт, 1 семестр

1. Информация. Данные. Информатика как наука, технология, индустрия.
2. Системы счисления.
3. Единицы измерения информации.
4. Файлы. Форматы файлов.

5. Устройство персонального компьютера.
6. История возникновения вычислительных машин.
7. Поколения ЭВМ.
8. Принципы фон Неймана.
9. Центральный Процессор
10. Способ структурной организации ЭВМ
11. Память компьютера. Основная память. Разновидность (адресная, стековая, ассоциативная, кеш). Представление информации в ОП.
12. Основные понятия алгебры логики. Логические операции. Построение логических схем.
13. Последовательностные функциональные узлы. Триггеры. Регистры и счетчики.
14. Комбинационные схемы, понятие об интегральной схеме.
15. Вспомогательная память. Магнитные диски. Дискеты. Оптические (лазерные) CD и DVD диски. Flash-память.
16. Управление вводом-выводом. Магистрально-модульный принцип построения компьютера.
17. Система прерываний.
18. Алгоритмы.
19. Тестирование программ.
20. Приведите общую структуру СОО. Охарактеризуйте ее уровни? Сформулируйте понятие интерфейса.
21. В чем состоит назначение виртуальной машины? Что является интерфейсом виртуальной машины? Установите связь виртуальной и физической машины.
22. Перечислите задачи, решаемые операционной системой.
23. Перечислите виды ресурсов СОО. Какие функции по их управлению выполняет ОС?
24. По каким характеристикам можно классифицировать ОС и оценивать ее эффективность?
25. В чем отличие истинного и кажущегося распараллеливания?
26. Какова аппаратная основа истинного распараллеливания?
27. В чем состоит задача планирования в многопрограммном режиме выполнения пакета?
28. Перечислите достоинства и недостатки пакетного режима и режима разделения времени.
29. Какова основная задача режима реального времени?
30. Приведите схему общей структуры ОС. Какие виды интерфейса предоставляет ОС пользователям?
31. В чем назначение привилегированного режима работы ядра? В любом ли типе ОС необходим привилегированный режим?
32. Зачем и как взаимодействуют с ядром прикладные программы?
33. Поясните термины «задача», «процесс», «поток». В каком случае понятие потока становится излишним?
34. Какая структурная единица - процесс или поток, требует защиты?
35. В чем состоят процессы планирования и диспетчеризации процессов и потоков?

36. Перечислите возможные состояния потока.
  37. Определите понятие прерывания. В чем различие между использованием прерывания и организацией программного ветвления?
  38. Приведите классификацию прерываний и схему их обработки.
  39. Сформулируйте цели синхронизации процессов и потоков. Опишите основные средства синхронизации.
  40. Определите понятия гонки и тупика. В чем отличие этих понятий?
  41. В чем отличие в использовании свопинга и виртуальной памяти?
  42. В чем отличие сегментации от страничной организации памяти?
  43. На каких механизмах и условиях основан процесс преобразования виртуального адреса в физический?
  44. Приведите схему сегментно-страничного распределения памяти.
  45. Перечислите задачи, решаемые ОС, по управлению файлами и устройствами.
  46. Приведите уровни модели подсистемы ввода-вывода.
  47. Какого назначения буферизация при выполнении операций обмена?
  48. Какого назначения механизма кэширования данных?
  49. В чем отличие в механизмах прямого доступа к внешней памяти магнитного диска и произвольного доступа к оперативной памяти?
  50. Какие составляющие включает организация файловой системы на логическом уровне?
  51. Опишите организацию физического уровня файловой системы с использованием FAT.
  52. Перечислите операции с файлами, доступные пользователям. В чем назначение операций открытия и закрытия файлов?
  53. Вредоносные программы. Разновидности, способы действия.
  54. Антивирусные программы и комплексы.
- Компьютерные сети. Назначение и Классификация сетей. Базовые принципы организации сети. Архитектура Клиент-Сервер.

## **Вопросы к экзамену, 2 семестр**

### Глава 1

1. Интуитивное определение алгоритма.
2. Алгоритм Евклида.
3. Что такое алгоритмически неразрешимая проблема.
4. Что значит доказать алгоритмическую неразрешимость.
5. Необходимость уточнения понятия алгоритма.
6. Алгоритмическая система.
7. Отличие алгоритмов поведения от вычислительных.
8. Свойства алгоритмов.
9. Свойство конструктивности алгоритма.
10. Свойство конечности алгоритма.
11. Свойство дискретности и элементарности алгоритма.
12. Свойство результативности алгоритма.
13. Свойство детерминированности алгоритма.
14. Свойство массовости алгоритма.
15. Определение алфавита.



16. Слово в алфавите.
17. Конкатенация слов.
18. Подслово.
19. Алфавитный оператор.
20. Область определения АО.
21. Полностью и частично определенные АО.
22. Однозначные и многозначные АО.
23. Определение алгоритма (через АО).
24. Равенство двух АО.
25. Равенство и эквивалентность алгоритмов.

## Глава 2

26. На каких элементарных действиях основано задание НАМ.
27. Как задается система подстановок НАМ.
28. Определение НАМ.
29. Порядок применения подстановок НАМ.
30. Условия останова НАМ.
31. Когда говорят, что НАМ неприменим к слову  $\square$ ?
32. Дедуктивная цепочка.
33. Особенности подстановки  $\square \square \square$ .
34. Принцип нормализации.
35. Значение принципа нормализации.
36. Что такое композиция алгоритмов?
37. Композиция НАМ: суперпозиция.
38. Теорема суперпозиции.
39. Композиция НАМ: объединение.
40. Теорема объединения.
41. Композиция НАМ: разветвление.
42. Теорема разветвления.
43. Композиция НАМ: повторение.
44. Теорема повторения.
45. Зачем нужны композиции алгоритмов?
46. Что такое универсальный НАМ?
47. Теорема об универсальном НАМ.

## Глава 3

48. Описание МТ.
49. Ситуация МТ.
50. Конфигурация МТ.
51. Какие действия может выполнить МТ в один момент времени?
52. Команда МТ.
53. Программа МТ.
54. Заключительная команда МТ.
55. Свойство детерминированности программы Мт.
56. Формальное определение МТ.
57. Таблица поведения МТ.
58. Диаграмма переходов МТ.
59. Гипотеза Тьюринга.

60. Значение гипотезы Тьюринга.
61. Назначение композиций НАМ.
62. МТ: последовательная композиция.
63. МТ: параллельная композиция.
64. МТ: композиция-разветвление (общий смысл).
65. МТ: композиция-разветвление, вспомогательная машина  $Z'$ .
66. МТ: композиция-разветвление, вспомогательная машина  $V$ .
67. Построение МТ  $W$  – композиции-разветвления  $X, Y, Z$ .
68. МТ: композиция-итерация.
69. Назначение универсальной МТ.
70. Теорема об универсальной МТ.
71. Исходные данные для универсальной МТ.
72. В чём состоит проблема распознавания применимости.
73. Доказать, что проблема распознавания самоприменимости алгоритмически неразрешима.
74. Машина Поста – отличие от МТ.
75. Команды МП.
76. Возможные ситуации при работе МП.

#### Глава 4

77. Что такое функция.
78. Вычислимая функция.
79. Арифметическая функция.
80. Элементарные арифметические функции.
81. Что такое оператор.
82. Определение оператора суперпозиции.
83. Общее понятие рекурсии.
84. Определение оператора примитивной рекурсии.
85. Примитивно-рекурсивные функции.
86. Какова область определения примитивно-рекурсивной функции.
87. Определение оператора минимизации.
88. Частично-рекурсивные функции.
89. Общерекурсивные функции.
90. Тезис Чёрча.
91. Значение тезиса Чёрча.
92. Теорема Маркова.
93. Теорема Тьюринга.
94. Теорема Детловса.
95. Соотношение трёх алгоритмических систем между собой и с интуитивным

понятием

алгоритма.

#### Глава 5

96. Алгоритм Дейкстры построения ПОЛИЗ простого арифметического выражения.
97. Требования к приоритету знака  $=$ .
98. ПОЛИЗ операции присваивания.
99. Формула вычисления адреса элемента одномерного массива.
100. Основные элементы языка.

101. Формула вычисления адреса элемента двумерного массива.
  102. Что такое лексема, примеры лексем.
  103. Операция АЭМ, её особенности.
  104. Что такое синтаксис.
  105. Дополнение к алгоритму Дейкстры по обработке [].
  106. Что такое семантика.
  107. Требования к приоритету знака ].
  108. Какие элементы языка изменяются при трансляции.
  109. Особенности трансляции операции АЭМ.
  110. Основные этапы трансляции.
  111. ПОЛИЗ составного оператора.
  112. Различие между прямыми и синтаксическими методами трансляции.
  113. Статические и динамические деревья.
  114. Построение дерева арифметического выражения.
  115. Операции УПЛ и БП.
  116. Правило левого обхода дерева.
  117. Дерево полного условного оператора.
  118. Свойства ПОЛИЗ.
  119. Дерево неполного условного оператора.
  120. Правило вычисления ПОЛИЗ.
  121. Дополнение к алгоритму Дейкстры по обработке знаков if, else, ), ; .
  122. Правила использования рабочих переменных при трансляции и вычислении ПОЛИЗ.
  123. Особенности трансляции операций УПЛ и БП.
  124. Что такое стек.
  125. Дерево оператора цикла while.
  126. Алгоритм вычисления ПОЛИЗ с использованием стека операндов.
  127. Дополнение к алгоритму Дейкстры по переводу оператора while.
- Глава 6
128. Что такое метаязык?
  129. Вид правил в грамматиках контекстно-зависимых языков.
  130. Вид правил в грамматиках контекстно-свободных языков.
  131. Задача и две стратегии синтаксического анализа.
  132. Определение формальной грамматики.
  133. Леворекурсивные и праворекурсивные грамматики.
  134. Грамматики класса 3 (автоматные).
  135. Алгоритм работы нисходящего распознавателя для грамматик класса 2.
  136. Что такое БНФ?
  137. Восходящий анализ.
  138. Определение:  $x \Rightarrow y$  (прямо порождает).
  139. Что такое основа (в алгоритмах восходящего анализа)?
  140. Определение:  $x \Rightarrow^* y$  (порождает).
  141. Задача лексического анализа.
  142. Что такое вывод?
  143. Что такое дескриптор?
  144. Определение формального языка.

145. Два типа таблиц лексем.
146. Как строится дерево вывода?
147. Почему этап лексического анализа обычно выделяется в отдельный блок транслятора?
148. Что такое разбор?
149. Диаграммы состояний для грамматик класса 3.
150. Определение:  $y \sqsubseteq x$  (прямо приводима).
151. Правило анализа предложения по диаграмме состояний.
152. Определение:  $y \sqsubseteq \sqsubseteq x$  (приводима).
153. Определение конечного автомата.
154. Что такое канонический разбор?
155. Алгоритм работы конечного автомата.
156. Чем отличаются дерево разбора и дерево вывода?
157. Правило анализа предложения по конечному автомату.
158. Классификация языков по Хомскому.
159. Что такое сканер?

Для получения зачёта по практике студенту необходимо выполнить и сдать все перечисленных выше задания. За каждое задание он получает максимально 4 балла.

#### **4. Методические материалы, определяющие процедуры оценивания образовательных результатов обучения**

4.1. Методические материалы для оценки текущего контроля успеваемости по дисциплине.

##### **Критерии оценивания Заданий**

Критерии	0 баллов	1 балл	2 балла
Правильность алгоритма	Алгоритм реализован неправильно, задача не решена	Алгоритм реализован так, что задача решена частично.	Алгоритм реализован правильно, задача решена
Правильность реализации алгоритма	Программы нет	Программа работает, но составлена с некоторыми ошибками	Программа составлена хорошо, все варианты реализованы

4.2. Методические материалы для проведения промежуточной аттестации по дисциплине.

Студент получает оценку «Зачтено» по практике, если за время обучения он выполнил все задания хотя бы на 2 балла.

«Не зачтено» – есть хотя бы одно задание, по которому набрано менее 2 баллов.

Промежуточная аттестация по теоретическому материалу по дисциплине проводится в форме устного зачета в 1 семестре и устного экзамена во 2 семестре. К зачету и экзамену допускаются только студенты, успешно прошедшие текущие аттестации.

Каждый билет для устного зачёта и экзамена состоит из трех теоретических вопросов по темам дисциплины. Во время устного зачета с оценкой в качестве дополнительных вопросов преподаватель может задавать вопросы по темам, не вошедшим в билет.