

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

Факультет инновационных технологий

УТВЕРЖДАЮ:  
Руководитель ОПОП

 О.В. Вусович

« 30 » 08 2021 г.

Оценочные материалы  
текущего контроля и промежуточной аттестации по дисциплине

**Информатика и программирование**

по направлению подготовки

**27.03.05 Инноватика**

Направленность (профиль) подготовки:  
**Управление инновациями и внедрение технологий**

Форма обучения  
**Очная**

Квалификация  
**Бакалавр**

## 1. Планируемые результаты освоения дисциплины

Результаты освоения дисциплины (индикатор достижения компетенции)	Планируемые образовательные результаты (ОР) обучения по дисциплине
ИОПК-7.1 Понимает принцип работы современных информационных технологий.	ОР 7.1.1 Объясняет принципы современных информационных технологий ОР 7.1.2 Классифицирует информационные технологии по различным критериям
ИОПК-10.1 Владеет методами формализации и алгоритмизации задач, а также знает типовые алгоритмы для решения практических задач цифровизации.	ОР 10.1.1 Использует методы формализации и алгоритмизации для решения задач ОР 10.1.2 Воспроизводит типовые алгоритмы для решения практических задач
ИОПК-10.2 Знает и способен применять современные среды разработки для практического применения.	ОР 10.2.1 Демонстрирует навыки работы с современными средами разработки ОР 10.2.2 Использует современные среды разработки для решения практических задач

## 2. Этапы достижения образовательных результатов в процессе освоения дисциплины

№	Разделы и(или) темы дисциплин	Образовательные результаты	Формы текущего контроля и промежуточной аттестации
1.	Тема 1. Введение в информатику и программирование.	ОР 7.1.1 ОР 7.1.2	Текущий контроль: Контроль посещаемости  Промежуточная аттестация: Зачет
2.	Тема 2. Ознакомление с операционной системой Linux и основными командами в терминале	ОР 7.1.2 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Контроль посещаемости  Промежуточная аттестация: Зачет
3.	Тема 3. Часто используемые команды в терминале Linux. Bash-скрипты.	ОР 7.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
4.	Тема 4. Введение в программирование на Python. Основы синтаксиса Python.	ОР 7.1.1 ОР 7.1.2 ОР 10.2.1	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
5.	Тема 5. Основы работы с системой контроля версий. Целочисленная арифметика в Python.	ОР 7.1.1 ОР 10.2.1	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа

			Промежуточная аттестация: Зачет
6.	Тема 6. Условный оператор	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
7.	Тема 7. Вложенные и каскадные условия. Числовой тип данных.	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
8.	Тема 8. Циклы for и while	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
9.	Тема 9. Операторы break и continue. Вложенные циклы.	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
10.	Тема 10. Модули math, random и time.	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
11.	Тема 11. Работа со строковым типом данных.	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
12.	Тема 12. Списки и кортежи	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация:

			Зачет
13.	Тема 13. Наборы и словари	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
14.	Тема 14. Работа с файлами	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
15.	Тема 15. Основы работы с функциями	ОР 10.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
16.	Тема 16. Модули	ОР 10.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Зачет
17.	Тема 17. Расширенная работа с функциями	ОР 10.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Экзамен
18.	Тема 18. Основы ООП. Классы.	ОР 7.1.1 ОР 7.1.2 ОР 10.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Экзамен
19.	Тема 19. Расширенная работа с классами. Наследование.	ОР 7.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Экзамен
20.	Тема 20. Декомпозиция при	ОР 7.1.1	Текущий контроль:

	решении задач программирования. Архитектура кода.	ОР 10.1.1 ОР 10.2.1 ОР 10.2.2	Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Экзамен
21.	Тема 21. Алгоритмизация при решении задач программирования. Основные алгоритмы.	ОР 7.1.1 ОР 10.1.1 ОР 10.1.2 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости Контрольная работа  Промежуточная аттестация: Экзамен
22.	Тема 22. Библиотека Matplotlib	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
23.	Тема 23. Python MySQL	ОР 7.1.1 ОР 7.1.2 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
24.	Тема 24. Библиотека NumPy	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
25.	Тема 25. Библиотека Pandas	ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
26.	Тема 26. Работа с Web	ОР 7.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
27.	Тема 27. Использование системы контроля версий при разработке сложных программ	ОР 7.1.1 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен
28.	Тема 28. Работа с Web. Django.	ОР 10.1.1 ОР 10.2.1	Текущий контроль: Домашняя работа

		ОР 10.2.2	Контроль посещаемости Промежуточная аттестация: Экзамен
29.	Тема 29. Введение в машинное обучение с использованием Python.	ОР 7.1.1 ОР 10.1.1 ОР 10.1.2 ОР 10.2.1 ОР 10.2.2	Текущий контроль: Домашняя работа Контроль посещаемости  Промежуточная аттестация: Экзамен

### 3. Оценочные средства для проведения текущего контроля и методические материалы, определяющие процедуру их оценивания

Текущий контроль проводится в течение семестра с целью определения уровня усвоения обучающимися знаний, формирования умений и навыков, своевременного выявления преподавателем недостатков в подготовке обучающихся и принятия необходимых мер по ее корректировке, а также для совершенствования методики обучения, организации учебной работы, и фиксируется в форме контрольной точки не менее одного раза в семестр.

#### 3.1. Контрольная работа по теме 3 – Часто используемые команды в терминале Linux. Bash-скрипты.

Задача 1. Написать Bash-скрипт который переходит в заданную директорию и создает новый файл с именем «test». Если такой файл уже существует, скрипт должен сначала удалить существующий файл.

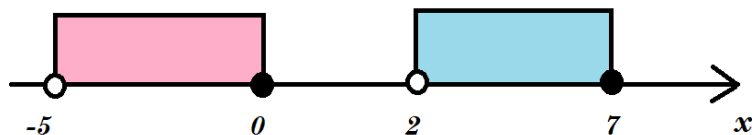
Задача 2. Написать Bash-скрипт который создает новую директорию с именем «testdir», после чего переходит в эту директорию и копирует в неё заданный файл.

Задача 3. Написать Bash-скрипт который ищет в заданном файле строки, содержащие текст: «ERROR»

#### 3.2. Примеры домашних работ

##### 3.2.1. Домашняя работа по теме 6. Условный оператор:

Даны два множества  $x$ . Написать программу, которая будет определять, принадлежат ли оба заданных числа одному из множеств.



##### Примечания:

1. Если граница множества закрашена (черная) – она включается во множество, если граница множества не закрашена (белая) – не включается.
2. Используйте логические операции, чтобы связывать условия (and, or)
3. Помните, что логическое умножение «and» имеет приоритет над логическим сложением «or»

**На вход** программе подаётся два целых числа.

**Программа должна вывести** текст: если оба числа принадлежат одному из

множеств – «Да», в других случаях – «Нет»

Пример входных/выходных данных:

Входные данные	Выходные данные
7 3	Да
0 -2	Да
-3 3	Нет
2 5	Нет

### 3.2.2. Домашняя работа по теме 8. Циклы for и while:

Напишите программу, которая считывает вводимые слова, пока не будет введено слово «PRINT».

На вход программе подаётся последовательность строк.

Программа должна вывести все эти строки с конкатенацией «id»

Пример входных/выходных данных:

Входные данные	Выходные данные
YOL TOOR SHUL PRINT	idYOL idTOOR idSHUL
12345 54321 PRINT	id12345 id54321

### 3.3. Пример варианта контрольной работы по темам 4 – 7:

Программа обрабатывает некоторый ключ действия. Гарантируется, что ключ действия — это строка, полученная конкатенацией (сложение строк) четырехзначного натурального числа  $N$  и одного из трех ключевых слов: *usr*, *clc*, *adm*, например: 1234clc.

Если ключ действия содержит ключевое слово *usr*, то программа должна предложить ввести натуральное число с комментарием «Подтвердите свой id-номер:»; если введенное число совпадает с изначально заданным числом  $N$ , то программа должна вывести текст «Пользователь подтвержден», иначе – «Неверный id-номер»

Если ключ действия содержит ключевое слово *clc*, то программа должна предложить ввести три любых числа, после чего выполнить с ними следующие операции:

1. перемножить модули всех трех чисел,
2. сложить все три числа,
3. определить наибольшее число,
4. определить наименьшее число,
5. разделить наибольшее число на наименьшее,
6. вычесть из наибольшего числа остальные числа,
7. вывести построчно результаты этих операций

Если ключ действия содержит ключевое слово *adm*, то программа должна предложить ввести строку. Если введенная строка не короче 5 символов, и имеет подстроку «.exe» или «.cmd», то программа выводит текст «Исполняемое действие», иначе – «Неисполняемое действие».

На вход программе подаётся натуральное четырехзначное число и строка.

Программа должна вывести текст, в зависимости от условий.

Пример входных/выходных данных:

Входные данные	Выходные данные
1234 usr >>> 1235	Ключ действия: 1234usr Подтвердите свой id-номер: >>> Неверный id-номер
5555 clc >>> 0.5 1.3 -7.0	Ключ действия: 5555clc Введите три любые числа: >>> Произведение модулей = 4.55 Сумма чисел = -5.2 Наибольшее = 1.3 Наименьшее = -7 Деление = -0.18571 Вычитание = 7.8
8800 adm >>> start.exe	Ключ действия: 8800adm Введите строку: >>> Исполняемое действие

Примечания:

1. В таблице входных/выходных данных символы «>>>» означают переход от вывода к вводу.
2. Готовая программа должна быть загружена на Git-Hub в репозиторий для контрольных работ.

#### 4. Оценочные средства для проведения промежуточной аттестации

**Зачет в первом семестре** выставляется по результатам проверки практических заданий и положительной оценки за тест по теоретическому материалу.

**Оценка за экзамен во втором семестре** выводится как среднее из двух оценок – оценка за практику (отлично, хорошо, удовлетворительно), оценка за тест по теории.

##### 4.1. Вопросы для подготовки к зачету в 1 семестре

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. Программа на линейный или условный алгоритм.

#### 4.2. Пречень тем к вопросам экзаменационного теста во 2 семестре

1. Приведите общую структуру СОО. Охарактеризуйте ее уровни? Сформулируйте понятие интерфейса.
2. В чем состоит назначение виртуальной машины? Что является интерфейсом виртуальной машины? Установите связь виртуальной и физической машины.
3. Перечислите задачи, решаемые операционной системой.
4. Перечислите виды ресурсов СОО. Какие функции по их управлению выполняет ОС?
5. По каким характеристикам можно классифицировать ОС и оценивать ее эффективность?
6. В чем отличие **истинного** и **кажущегося** распараллеливания?
7. Какова аппаратная основа **истинного** распараллеливания?
8. В чем состоит задача планирования в многопрограммном режиме выполнения пакета?
9. Перечислите достоинства и недостатки пакетного режима и режима разделения времени.
10. Какова основная задача режима реального времени?
11. Приведите схему общей структуры ОС. Какие виды интерфейса предоставляет ОС пользователям?
12. В чем назначение привилегированного режима работы ядра? В любом ли типе ОС необходим привилегированный режим?
13. Зачем и как взаимодействуют с ядром прикладные программы?
14. Поясните термины «задача», «процесс», «поток». В каком случае понятие потока становится излишним?
15. Какая структурная единица - процесс или поток, требует защиты?
16. В чем состоят процессы планирования и диспетчеризации процессов и потоков?
17. Перечислите возможные состояния потока.
18. Определите понятие **прерывания**. В чем различие между использованием прерывания и организацией программного ветвления?
19. Приведите классификацию прерываний и схему их обработки.
20. Сформулируйте цели синхронизации процессов и потоков. Опишите основные средства синхронизации.
21. Определите понятия **гонки** и **тупика**. В чем отличие этих понятий?
22. В чем отличие в использовании **свопинга** и **виртуальной памяти**?
23. В чем отличие **сегментации** от **страничной организации** памяти?
24. На каких механизмах и условиях основан процесс преобразования виртуального адреса в физический?
25. Приведите схему сегментно-страничного распределения памяти.
26. Перечислите задачи, решаемые ОС, по управлению файлами и устройствами.
27. Приведите уровни модели подсистемы ввода-вывода.
28. Какого назначения **буферизации** при выполнении операций обмена?
29. Какого назначения механизма **кэширования** данных?
30. В чем отличие в механизмах прямого доступа к внешней памяти магнитного диска и произвольного доступа к оперативной памяти?

31. Какие составляющие включает организация файловой системы на логическом уровне?
32. Опишите организацию физического уровня файловой системы с использованием FAT.
33. Перечислите операции с файлами, доступные пользователям. В чем назначение операций открытия и закрытия файлов?
34. Вредоносные программы. Разновидности, способы действия.
35. Антивирусные программы и комплексы.
36. Компьютерные сети. Назначение и Классификация сетей. Базовые принципы организации сети. Архитектура Клиент-Сервер.
37. Программа на массивы.

### **4.3 Критерии оценивания**

Тест состоит из 40 вопросов.

Тест оценивается следующим образом:

Отлично – от 97% до 100% правильных ответов

Хорошо – от 90% до 96% правильных ответов

Удовлетворительно – от 70% до 89% правильных ответов

Неудовлетворительно – менее 69% правильных ответов