

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

Факультет инновационных технологий

УТВЕРЖДАЮ:

Декан



С. В. Шидловский

« 16 » 05 2023 г.

Рабочая программа дисциплины

**Информатика и программирование**

по направлению подготовки

**27.03.05 Инноватика**

Направленность (профиль) подготовки :

**Управление инновациями в наукоемких технологиях**

Форма обучения

**Очная**

Квалификация

**Бакалавр**

Год приема

**2023**

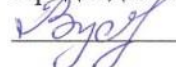
Код дисциплины в учебном плане: Б1.О.13

СОГЛАСОВАНО:

Руководитель ОПОП

 О.В. Вусович

Председатель УМК

 О.В. Вусович

## **1. Цель и планируемые результаты освоения дисциплины**

Целью освоения дисциплины является формирование следующих компетенций:

- ОПК-7 – Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности;
- ОПК-10 – Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК-7.1 Понимает принцип работы современных информационных технологий.

ИОПК-10.1 Владеет методами формализации и алгоритмизации задач, а также знает типовые алгоритмы для решения практических задач цифровизации.

ИОПК-10.2 Знает и способен применять современные среды разработки для практического применения.

## **2. Задачи освоения дисциплины**

– Изучить устройство персонального компьютера, получить основы работы с алгеброй логики, узнать архитектуру операционной системы и типы программного обеспечения персонального компьютера.

– Научиться разрабатывать алгоритмы и создавать программы, применять программирование в своей профессиональной деятельности.

– Знать основы построения и использования компьютерных сетей, а также основы компьютерной безопасности.

– Научиться работать в операционных системах Linux на уровне уверенного пользователя.

– Освоить терминал Linux для решения пользовательских задач.

– Понимать принцип системы контроля версий при разработке ПО.

– Овладеть навыками работы с системами контроля версий.

– Сформировать понимание важности качества кода при разработке ПО.

– Научиться работать с интегрированными средами разработки (IDE).

– Освоить синтаксис языка программирования Python.

– Сформировать понимание принципов формализации, декомпозиции и алгоритмизации при решении задач.

– Научиться решать практические задачи с использованием как стандартных, так и расширенных возможностей средств разработки.

## **3. Место дисциплины в структуре образовательной программы**

Дисциплина относится к обязательной части образовательной программы.

## **4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине**

Первый семестр, зачет

Второй семестр, зачет с оценкой

## **5. Входные требования для освоения дисциплины**

Для успешного освоения дисциплины требуются компетенции, сформированные в ходе освоения образовательных программ предшествующего уровня образования.

## **6. Язык реализации**

Русский

## **7. Объем дисциплины**

Общая трудоемкость дисциплины составляет 6 з.е., 216 часов, из которых:

-лекции: 48 ч.

-практические занятия: 66 ч.

Объем самостоятельной работы студента определен учебным планом.

## **8. Содержание дисциплины, структурированное по темам**

### **Темы лекционных занятий:**

Тема 1. Информатика. Информация. Данные. Автоматизация информации. Единицы измерения информации. Файл

Понятие информации. Основные определения. Информатика как наука, как технология и как индустрия. Автоматизация информации. Единицы измерения информации. Файл.

Тема 2. Системы счисления

Основные понятия систем счисления. Перевод чисел из одной системы счисления в другую. Алгоритмы перевода

Тема 3. История возникновения вычислительных машин

История возникновения вычислительной техники. Поколения ЭВМ.

Тема 4. Введение в архитектуру ЭВМ. Устройство центрального процессора.

Память компьютера.

Структура и принцип функционирования ЭВМ. Процессор. Способ структурной организации ЭВМ.

Основная память. Разновидность, структура. Организация ОП. Адреса памяти. Представление информации в ОП. Магнитные диски. Дискеты. Магнитные ленты. Оптические (лазерные) CD и DVD диски. Оптические (лазерные) CD и DVD диски. Flash-память.

Тема 5. Основные понятия алгебры логики. Построение логических схем.

Последовательностные функциональные узлы. Триггеры

Логика. Высказывания. Булева алгебра. Логические элементы. Построение логических схем. Последовательностные функциональные узлы. Триггеры. Микросхемы.

Тема 6. Управление вводом-выводом. Магистрально-модульный принцип построения компьютера

Магистраль. Прерывания. Виды прерываний.

Тема 7. Основы алгоритмизации и программирования.

Линейный алгоритм. Операции взятия остатка от деления, получение целой части от деления. Оператор ветвления. Оператор выбора. Цикл-счетчик. Работа с вложенными циклами. Цикл «условие».

Тема 8. Назначение и характеристики операционных систем (ОС).

Назначение и состав системы обработки информации. Понятие виртуальной машины. Состав системы обработки информации. Назначение и функции системных программ. Компоненты системного программного обеспечения. Задачи, решаемые

операционной системой. Операционная система как виртуальная машина. Оценка деятельности операционной системы. Классификация операционных систем. Эволюция и режимы функционирования ОС.

#### Тема 9. Архитектура (структура) ОС

Ядро и вспомогательные модули. Ядро в привилегированном режиме. Многослойная структура операционной системы. Средства аппаратной поддержки ОС. Менеджеры ресурсов. Функциональные компоненты ОС. Управление процессами. Состояния потока. Мультипрограммирование на основе прерываний. Синхронизация процессов и потоков. Управление памятью. Ввод-вывод и файловая структура.

#### Тема 10. Компьютерные сети

Назначение и классификация сетей. Основные определения. Базовые требования, определяющие архитектуру компьютерных сетей. Уровни взаимодействия. Архитектура клиент-сервер. Глобальная сеть Интернет

#### Тема 11. Программное обеспечение персонального компьютера

Виды программного обеспечения персонального компьютера.

#### Тема 12. Вирусы и антивирусы

Вирусы. Классификация. Наиболее известные вирусы. Антивирусы. Основные подходы к обеспечению безопасности. Фаерволы.

#### **Темы практических занятий:**

Тема 1. Введение в информатику и программирование.

Рассказывается о целях и задачах курса. Раскрывается место курса в контексте инновационной деятельности. Приводится информация об изучаемом технологическом стеке, а также сопутствующие изучению литературные источники.

Тема 2. Ознакомление с операционной системой Linux и основными командами в терминале

Рассказывается о семействе операционных систем Linux, различных дистрибутивах, их сходствах и различиях друг с другом и другими ОС, а также их достоинства и недостатки. Знакомство с интерфейсом дистрибутива Linux. Запуск и работа с терминалом Linux в контексте основных пользовательских команд.

Тема 3. Часто используемые команды в терминале Linux. Bash-скрипты.

Приводится информация о наиболее используемых и полезных командах в терминале Linux. Рассказывается о скриптах, как о методе автоматизации для постоянных и рутинных процессов. Написание bash-скриптов с использованием изученных команд терминала Linux.

Тема 4. Введение в программирование на Python. Основы синтаксиса Python.

Рассказывается о языке программирования Python, включая его основные характеристики, преимущества, недостатки, область применения. Знакомство с интегрированной средой разработки (IDE). Изучение команд ввода и вывода, а также целочисленного и строкового типов данных.

Тема 5. Основы работы с системой контроля версий. Целочисленная арифметика в Python.

Рассказывается о системах контроля версий, их применении в контексте разработки ПО. Приводится пример создания репозитория и загрузки в репозиторий файла

программы. Рассматриваются возможности Python в области целочисленной арифметики, включая все операторы. Создается простая программа-калькулятор для закрепления материала и загружается в собственный репозиторий в системе контроля версий.

#### Тема 6. Условный оператор

Изучается синтаксис и применимость условного оператора в Python. Приводится основная информация о логических операциях для применения в условном операторе. Решается несколько задач для закрепления материала.

#### Тема 7. Вложенные и каскадные условия. Числовой тип данных.

Изучается синтаксис и применимость вложенных условий в Python. Изучается синтаксис и применимость каскадных условий в Python. Приводятся примеры использования как вложенных, так и каскадных условий. Изучается числовой тип данных (числа с плавающей точкой), его отличия от целочисленного типа. Решается несколько задач для закрепления материала.

#### Тема 8. Циклы for и while

Изучается синтаксис и применимость цикла for в Python. Приводятся частые сценарии использования цикла for. Изучается синтаксис и применимость цикла while в Python. Приводятся частые сценарии использования цикла while. Решается несколько задач для закрепления материала.

#### Тема 9. Операторы break и continue. Вложенные циклы.

Изучается синтаксис и применимость операторов break и continue в Python. Рассматриваются бесконечные циклы и их значимость на практике. Изучается синтаксис и применимость вложенных циклов в Python. Решается несколько задач для закрепления материала.

#### Тема 10. Модули math, random и time.

Изучается синтаксис подключения модулей в Python. Рассматриваются возможности модулей math, random, time. Решается несколько задач для закрепления материала.

#### Тема 11. Работа со строковым типом данных.

Изучается индексация и итерирование строк. Рассматривается работа со срезами строк и возможности изменения символов строки. Изучаются синтаксис и применимость методов строк.

#### Тема 12. Списки и кортежи

Изучается синтаксис и применимость списков. Рассматривается работа со списками в контексте индексов, срезов, конкатенации. Изучаются методы списков. Изучается синтаксис и применимость кортежей. Приводятся примеры работы с кортежами и списками. Решается несколько задач для закрепления материала.

#### Тема 13. Наборы и словари

Изучается синтаксис и применимость словарей. Рассматривается работа со словарями в Python. Изучаются методы словарей. Изучается синтаксис и применимость наборов. Приводятся примеры работы с наборами и словарями. Решается несколько задач для закрепления материала.

#### Тема 14. Работа с файлами

Изучается синтаксис для работы с файлами. Рассматриваются понятия файла, файлового ввода и вывода. Изучается работа с файлами, методы доступа к файлам и основные атрибуты файлов. Решается несколько задач для закрепления материала.

#### Тема 15. Основы работы с функциями

Изучается синтаксис и применимость функций. Рассматриваются понятия пространства имен и области видимости. Приводятся примеры работы с функциями. Решается несколько задач для закрепления материала.

#### Тема 16. Модули

Изучается синтаксис и применимость модулей. Приводится обоснование переиспользования кода в контексте декомпозиции и проектного структурирования. Изучаются встроенные инструменты работы с модулями. Приводятся примеры работы с модулями. Решается несколько задач для закрепления материала.

#### Тема 17. Расширенная работа с функциями

Изучается синтаксис и применимость лямбда-функций, рекурсивных функций, встроенных функций, функций-генераторов, декораторов. Приводятся примеры расширенной работы с функциями. Решается несколько задач для закрепления материала.

#### Тема 18. Основы ООП. Классы.

Рассказывается об основных парадигмах программирования: императивном, структурном, объектно-ориентированном, функциональном, логическом. Приводятся основные понятия и особенности объектно-ориентированного программирования (ООП). Раскрывается значимость класса – как шаблона для создания объекта. Изучается синтаксис и применимость классов. Приводятся примеры работы с классами. Решается несколько задач для закрепления материала.

#### Тема 19. Расширенная работа с классами. Наследование.

Рассказывается о наследовании, как понятии ООП и месте наследования в контексте программного класса. Изучается синтаксис и применимость дочерних классов. Приводятся примеры работы с наследованием классов. Решается несколько задач для закрепления материала.

#### Тема 20. Декомпозиция при решении задач программирования. Архитектура кода.

Рассказывается о применении принципов формализации и декомпозиции в контексте решения задач программирования. Раскрываются принципы декомпозиции задач программирования и соответствующее структурирование программного кода. Приводятся примеры архитектуры ПО. Решается несколько задач для закрепления материала.

Тема 21. Алгоритмизация при решении задач программирования. Основные алгоритмы.

Рассказывается об алгоритмизации, как шаблоне для решения задач программирования. Рассматриваются примеры алгоритмов при решении типовых задач. Изучаются алгоритмы для сортировки (пузырьком, быстрая сортировка) и поиска (двоичный поиск, поиск в глубину). Решается несколько задач для закрепления материала.

Тема 22. Библиотека Matplotlib.

Изучается работа с библиотекой Matplotlib. Рассматриваются вопросы построения графиков и диаграмм посредством Matplotlib. Решается несколько задач для закрепления материала.

Тема 23. Python MySQL.

Изучается работа с базами данных (MySQL). Рассматриваются вопросы создания и работы с базами данных с использованием Python и MySQL. Решается несколько задач для закрепления материала.

Тема 24. Библиотека NumPy.

Изучается работа с библиотекой NumPy. Рассматриваются вопросы базовых функций библиотеки, функций для работы со случайным распределением, массивами и матрицами. Решается несколько задач для закрепления материала.

Тема 25. Библиотека Pandas.

Изучается работа с библиотекой Pandas. Рассматриваются возможности использования библиотеки Pandas для работы с наборами данных: анализ, очистка, изучение, формирование выводов. Решается несколько задач для закрепления материала.

Тема 26. Работа с Web.

Рассказывается об основных понятиях клиент-серверной архитектуры Web-приложений, запросах, ответах, API. Изучается работа с модулями cgi, html, json. Решается несколько задач для закрепления материала.

Тема 27. Использование системы контроля версий при разработке сложных программ.

Рассказывается о системах контроля версий в контексте использования при разработке сложных программ. Раскрываются понятия версионности, ветвления, параллельного программирования. Приводятся примеры работы в системе контроля версий при разработке сложных проектов.

Тема 28. Работа с Web. Django.

Изучается работа с фреймворком Django. Рассматриваются вопросы установки фреймворка, архитектуры проекта Django и работы с Django. В качестве практической задачи разрабатывается простой сайт (визитка) или веб-приложение.

Тема 29. Введение в машинное обучение с использованием Python.

Рассказываются основные понятия машинного обучения. Рассматриваются простейшие примеры работы с данными, включая анализ и экспертизу. В качестве практической задачи разрабатывается проект для анализа некоторых данных, например валютного курса.

## **9. Текущий контроль по дисциплине**

Текущий контроль по дисциплине проводится путем контроля посещаемости, проведения контрольных работ, выполнения домашних заданий, и фиксируется в форме контрольной точки не менее одного раза в семестр.

## **10. Порядок проведения и критерии оценивания промежуточной аттестации**

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

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

Тест состоит из 40 вопросов.

Примерный перечень тем к вопросам теста:

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.

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

Оценка за практику ставится следующим образом:

Тест оценивается следующим образом:

Отлично – все практические задания сданы вовремя и без ошибок,

Хорошо – все практические задания сданы до экзамена, есть незначительные ошибки,

Удовлетворительно – сдано не менее 80% практических заданий,

Неудовлетворительно – сдано менее 79% практических заданий.

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

Примерный перечень тем к вопросам теста:

1. Приведите общую структуру СОИ. 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 СОИ. 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 **тустика**. 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.

## 11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle» - <https://moodle.tsu.ru/course/view.php?id=881>  
<https://moodle.tsu.ru/course/view.php?id=2798>

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

в) Практические задания состоят из написания программ для решения поставленной задачи индивидуально. Для выполнения практической работы необходимо:

1. Прочитать и понять постановку задачи.

2. Создать программу, используя указанный или наиболее подходящий для этого алгоритмический язык.

3. Представить работу программы.

4. Сдать готовую работу преподавателю, объяснив код и показав работу программы.

д) На самостоятельную работу выносятся окончательное выполнение практических заданий.

## 12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

1. Бизли Д. Python. Книга рецептов / Бизли Д., Джонс Б.К.. – М.: ДМК Пресс, 2019. - 646 с. URL: <https://e.lanbook.com/book/131723>.

2. Гниденко И.Г. Технологии и методы программирования: Учебное пособие для вузов / Гниденко И.Г., Павлов Ф. Ф., Федоров Д. Ю.. – М.: Юрайт, 2022. - 235 с. URL: <https://urait.ru/bcode/489920>

3. Гостев И.М. Операционные системы: Учебник и практикум для вузов / Гостев И. – М.: Юрайт, 2022. - 164 с. URL: <https://urait.ru/bcode/490157>. URL: <https://urait.ru/book/cover/6AD79D06-CB76-4BDC-954B-30AF73EB2F3E>

4. Гуриков С.Р. Основы алгоритмизации и программирования на Python: Учебное пособие / С.Р. Гуриков. – М.: ООО "Научно-издательский центр ИНФРА-М", 2022. - 343 с.. URL: <http://znanium.com/catalog/document?id=379975>

5. Гэддис Т. Начинаем программировать на Python : пер. с англ. / Тони Гэддис. - 4-е изд.. - СПб: БХВ-Петербург, 2019. - 745 с.

6. Жуков Р.А. Язык программирования Python: практикум: Учебное пособие / Р.А. Жуков. – М.: ООО "Научно-издательский центр ИНФРА-М", 2021. - 216 с.. URL: <http://znanium.com/catalog/document?id=376046>

7. Новожилов О.П. Архитектура ЭВМ и систем в 2 ч. Часть 2 : Учебное пособие для вузов / Новожилов О. П.. – М.: Юрайт, 2022. - 246 с. URL: <https://urait.ru/bcode/494315>. URL: <https://urait.ru/book/cover/BBB21D8D-3638-4B31-A657-8AEF14762930>

8. Толстобров А.П. Архитектура ЭВМ: Учебное пособие для вузов / Толстобров А. П. – М.: Юрайт, 2022. - 154 с. URL: <https://urait.ru/bcode/496167>. URL: <https://urait.ru/book/cover/8B735316-BCF9-4E5B-8B90-2BCDDE722F97>

9. Федоров Д. Ю. Программирование на языке высокого уровня Python: Учебное пособие для вузов / Д.Ю. Федоров. – М.: Юрайт, 2022. - 210 с. URL: <https://urait.ru/bcode/492920>

10. Чернышев С. А. Основы программирования на Python: Учебное пособие для вузов / С.А. Чернышев. – М.: Юрайт, 2022. - 286 с. URL: <https://urait.ru/bcode/496893>

б) дополнительная литература:

1. Воган Л. "Непрактичный" Python: занимательные проекты для тех, кто хочет поумнеть : пер. с англ. / Ли Воган. - СПб: БХВ-Петербург, 2021. - 463 с.
2. Головчинер М.Н. Введение в операционные системы: курс лекций / М. Н. Головчинер. - Томск, 2009. - 109 с. URL: <http://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000428977>
3. Доусон М. Програмируем на Python / Майкл Доусон; пер. с англ. В. Порицкий ; науч. ред. В. Керимов. - СПб: Питер, 2021. - 414 с.
4. Златопольский Д.М. Основы программирования на языке Python / Д.М. Златопольский. – М.: ДМК Пресс, 2018. - 396 с. URL: <https://e.lanbook.com/book/131683>
5. Новожилов О. П. Информатика: учебник для прикладного бакалавриата: Учебник / Новожилов О.П. - М: Издательство Юрайт, 2017. – 619 с. URL: <http://www.biblio-online.ru/book/FEE705BC-11CB-46EB-810E-2634A4DE5E46>
6. Шелудько В. Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули: Учебное пособие. - Ростов-на-Дону: Издательство Южного федерального университета (ЮФУ), 2017. - 107 с.. URL: <http://znanium.com/catalog/document?id=339835>
7. Шолле Ф. Глубокое обучение на Python / Франсуа Шолле; пер. с англ. А. Киселев. - СПб: Питер, 2019. - 397 с.

в) ресурсы сети Интернет:

- Python документация. <https://docs.python.org/3/>
- Документация Linux (зависит от дистрибутива, пример для Ubuntu): <https://help.ubuntu.ru/>
- Документация Git Hub. <https://docs.github.com/en>
- Roger Dudler. Git - the simple guide. <https://rogerdudler.github.io/git-guide/>

### 13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

Для проведения лекционных и практических занятий необходимо лицензионное обеспечение:

- OS Linux, Python 3.7 +, PyCharm IDE, Git Hub;
- публично доступные облачные технологии (Google Docs, Яндекс диск).

б) информационные справочные системы:

- Электронный каталог Научной библиотеки ТГУ – <http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>
- Электронная библиотека (репозиторий) ТГУ – <http://vital.lib.tsu.ru/vital/access/manager/Index>
- ЭБС Лань – <http://e.lanbook.com/>
- ЭБС Консультант студента – <http://www.studentlibrary.ru/>
- Образовательная платформа Юрайт – <https://urait.ru/>
- ЭБС ZNANIUM.com – <https://znanium.com/>
- ЭБС IPRbooks – <http://www.iprbookshop.ru/>

### 14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

Аудитории для проведения занятий лекционного и семинарского типа индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации в смешенном формате («Актру»).

### **15. Информация о разработчиках**

Самохина Светлана Ивановна, кандидат физико-математических наук, доцент, доцент кафедры Информационного обеспечения инновационной деятельности Факультета инновационных технологий;

Гимазов Руслан Уралович, кандидат технических наук, ассистент кафедры Информационного обеспечения инновационной деятельности Факультета инновационных технологий.