

* + - 1. Цели освоения дисциплины

**Цель дисциплины**:

* ознакомление с алгоритмами и процессами решения задач, с событийно-управляемым и параллельным программированием; с прикладными программными интерфейсами (АРI) и их применением;
* ознакомление с основными конструкциями программирования; основными структурами данных; и с объектно-ориентированным программированием;
* научить составлять алгоритмы линейной, разветвляющейся, циклической структур; пользоваться классическими алгоритмами; процедурным программированием, рекурсией; объектно-ориентированным программированием

Основные задачи дисциплины:

* знакомство с основными парадигмами и теоретическими основами программирования, терминологией объектно-ориентированного программирования;
* изучение типов данных;
* изучение линейного, ветвящегося и циклического алгоритмов;
* изучение способов и механизмов управления данными;
* практическое освоение реализации изученных алгоритмов;
* выработка навыков программирования и алгоритмизации с применением современных процедурных и объектно-ориентированных языков программирования.

.

* + - 1. Место дисциплины в структуре ОП

Дисциплина «Методы программирования» читается студентам 2 курса и относится к числу дисциплин базовой части Блока1 Дисциплины (Модули).

Для успешного усвоения дисциплины «Методы программирования» необходимо, чтобы студент владел знаниями, умениями и навыками, сформированными при изучении таких дисциплин, как:

* «Математический анализ»,
* «Информатика»,
* «Языки программирования»,
* «Аппаратные средства вычислительной техники»,
* «Интернет-технологии».

Параллельно с данной дисциплиной идет изучение таких взаимосвязанных с «Методами программирования» с дисциплин как:

* «Теория вероятностей и математическая статистика»
* «Дискретная математика»
* «Аналитическая геометрия»
* «Информационные технологии»

Дисциплина «Методы программирования» обеспечивает изучение следующих дисциплин:

* «Системы управления базами данных»
* «Дисциплины специализации»
* «Операционные системы»
* «Теория информации и кодирования»;
* «Java-программирование»

Дисциплина «Методы программирования» необходима для подготовка курсовых работ и дипломного проектирования

* + - 1. Компетенции обучающегося, формируемые в результате освоения

**дисциплины.**

Процесс изучения дисциплины направлен на формирование следующих компетенций:

|  |  |
| --- | --- |
| **Код компетенции** | **Компетенция** |
| ОПК-4 | способностью понимать значение информации в развитии современного общества, применять достижения информационных технологий для поиска и  обработки информации |
| ОПК-5 | способностью применять программные средства системного и прикладного  назначения, языки, методы и инструментальные средства программирования для решения профессиональных задач |

В результате освоения компетенций в рамках дисциплины обучающийся должен:

|  |  |
| --- | --- |
| **Код компетенции** | **Результаты обучения** |
| ОПК-4 | Знать:   * парадигмы и методологии программирования; * особенности наиболее распространенных объектно-ориентированных языков программирования; * базовые структуры данных; * основные комбинаторные и теоретико-графовые алгоритмы; * современные технологии программирования   Уметь:   * формализовать поставленную задачу; * работать с интегрированными средами разработки программного обеспечения; * применять в профессиональной деятельности современные объектно-ориентированные языки программирования; * профессионально решать задачи производственной и технологической деятельности с учетом современных достижений науки и техники; * оценивать качество готового программного обеспечения   Владеть:   * навыками постановки математических и информационных задач; * навыками описания алгоритмов решения поставленной задачи и разработки программного кода; * навыками тестирования программного обеспечения; разработки алгоритмов решения типовых профессиональных задач |
| ОПК–5 | Знать:   * современные методы программирования; * методы составления алгоритмов; * основные приемы реализации алгоритмов на языках программирования; * основные приемы программирования. * паттерны проектирования; * методы тестирования и отладки ПО. Уметь: * реализовывать на языке высокого уровня алгоритмы решения профессиональных задач, в том числе задач обработки потоков; * работать с интегрированной средой разработки программного обеспечения; * реализовывать на языке высокого уровня алгоритмы решения профессиональных задач, в том числе задач обработки потоков.   Владеть:   * навыками решения практических задач, * методами программирования, * навыками разработки, документирования, тестирования и отладки   программ |

Основные признаки проявленности формируемых компетенций в результате освоения дисциплины «Методы программирования» сведены в таблице.

Соответствие уровней освоения компетенции планируемым результатам обучения и критериям их оценивания

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Уровень освоения компетенции** | **Результат обучения** | | | **Результат обучения** | | | | |
| ОПК-4: Знать, уметь, владеть | | | ОПК-5: Знать, уметь, владеть | | | | |
| минимальный | Способен критическую  материала | дать оценку | собственную изучаемого | Владеет основными навыками работы с источниками и критической  литературой | | | | |
| Может соотнести основные идеи с современными проблемами | | | Способен  проблему процессами | в | представить ее связи | с | ключевую другими |
| Способен выделить характерный  авторский подход | | | Понимает специфику основных  рабочих категорий | | | | |
| базовый | Способен сравнивать концепции, аргументированно излагает материал | | | Свободно излагает материал, однако не демонстрирует навыков сравнения  основных идей и концепций | | | | |
| Аргументированно проводит сравнение концепций по заданной проблематике | | | Способен выделить и сравнить  концепции, но испытывает сложности с их практической привязкой | | | | |
| Способен  концепций в области | выделить заданной | специфику проблемной | Знает основные отличия концепций в заданной проблемной области | | | | |
| продвинутый | Способен грамотно обосновать собственную позицию относительно решения современных проблем в  заданной области | | | Видит источники современных проблем в заданной области анализа, владеет подходами к их решению | | | | |
| Свободно ориентируется в заданной области анализа. Понимает ее основания и умеет выделить  практическое значение заданной области | | | Выявляет основания заданной области анализа, понимает ее практическую ценность, однако испытывает  затруднения в описании сложных объектов анализа | | | | |
| Может дать критический анализ современным проблемам в заданной области анализа | | | Знает основное содержание современных научных идей в рабочей области анализа, способен их  сопоставить | | | | |

**Соответствие уровней освоения компетенции планируемым результатам обучения и критериям их оценивания**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Этап (уровень) освоения  компетенции | Основные признаки проявленности компетенции (дескрипторное описание уровня) | | | | |
| 1. | 2. | 3. | 4. | 5. |
| минимальный | не владеет | слабо ориентируется  в терминологии и содержании | Способен выделить основные идеи  текста, работает с критической литературой | Владеет основными навыками работы с источниками и критической литературой | Способен дать собственную  критическую оценку изучаемого материала |
| не умеет | не выделяет  основные идеи | Способен показать основную идею в  развитии | Способен представить ключевую проблему  в ее связи с другими процессами | Может соотнести основные идеи с  современными проблемами |
| не знает | допускает грубые ошибки | Знает основные рабочие категории, однако не ориентируется в их  специфике | Понимает специфику основных рабочих категорий | Способен выделить характерный авторский подход |
| базовый | не владеет | плохо ориентируется  в терминологии и содержании | Владеет приемами поиска и  систематизации, но не способен свободно изложить материал | Свободно излагает материал, однако не  демонстрирует навыков сравнения основных идей и концепций | Способен сравнивать концепции,  аргументированно излагает материал |
| не умеет | выделяет основные  идеи, но не видит проблем | Выделяет конкретную проблему, однако излишне упрощает ее | Способен выделить и сравнить концепции,  но испытывает сложности с их практической привязкой | Аргументированно проводит  сравнение концепций по заданной проблематике |
| не знает | допускает много ошибок | Может изложить основные рабочие категории | Знает основные отличия концепций в заданной проблемной области | Способен выделить специфику концепций в заданной проблемной  области |
| продвинутый | не владеет | ориентируется в терминологии и содержании | В общих чертах понимает основную идею, однако плохо связывает ее с существующей проблематикой | Видит источники современных проблем в заданной области анализа, владеет подходами к их решению | Способен грамотно обосновать собственную позицию  относительно решения  современных проблем в заданной области |
| не умеет | выделяет основные идеи, но не видит их в развитии | Может понять практическое назначение основной идеи, но затрудняется выявить ее основания | Выявляет основания заданной области анализа, понимает ее практическую ценность, однако испытывает затруднения в описании сложных объектов анализа | Свободно ориентируется в заданной области анализа. Понимает ее основания и умеет  выделить практическое значение заданной области |
| не знает | допускает ошибки при выделении рабочей области  анализа | Способен изложить основное содержание современных научных идей в рабочей области анализа | Знает основное содержание современных научных идей в рабочей области анализа, способен их сопоставить | Может дать критический анализ современным проблемам в заданной области анализа |

* + - 1. Структура и содержание дисциплины

Общая трудоемкость (объем) дисциплины (модуля) составляет 4 зачетных единиц (ЗЕ\*), 144 академических часа.

Год набора: 2020

|  |  |  |
| --- | --- | --- |
| **Объём дисциплины** | **Всего часов** | **4 семестр** |
| **Общая трудоёмкость дисциплины** | 144 | 144 |
| **Контактная работа обучающихся с преподавателям (по видам аудиторных учебных занятий) – всего:** | 56 | 56 |
| в том числе: |  |  |
| лекции | 28 | 14 |
| Лабораторные занятия | 28 | 28 |
| **Самостоятельная работа** (СРС) – всего: | 88 | 88 |
| курсовая работа |  |  |
| **Вид промежуточной аттестации (зачет/экзамен)** | Экзамен | Экзамен |

* + - * 1. Содержание разделов дисциплины

Методы программирования

Введение

Основные понятия и подходы. Приемы обеспечения технологичности программных продуктов. Разработка технического задания . Этапы развития технологии программирования. Модели жизненного цикла программного обеспечения. Понятия эффективности и технологичности программного обеспечения. Модули и их свойства. Средства описания структурных алгоритмов (псевдокоды, блок-схемы алгоритмов, Flow-формы, диаграммы Насси-Шнейдермана). Правила оформления программ.

Проектирование программного обеспечения при структурном подходе к программированию

Диаграммы переходов состояний. Функциональные диаграммы. Диаграммы потоков данных. Диаграммы отношений компонентов данных: диаграммы Джексона и скобочные диаграммы Орра, сетевая модель данных (Диаграммы «сущность-связь»). Структурная и функциональная схемы. Структурные карты Константайна.

Тестирование и отладка программных продуктов при структурном подходе.

Структурное тестирование. Функциональное тестирование. Классификация ошибок. Методы отладки программного обеспечения.

Проектирование программного обеспечения при объектно-ориентированном подходе к программированию

UML- стандартный язык описания разработки программных продуктов с использованием объектного подхода. Диаграммы вариантов использования. Диаграмма классов. Диаграмма последовательностей. Диаграмма деятельности. Диаграмма пакетов. Диаграмма кооперации. Диаграмма состояний объекта. Диаграмма компонентов. Диаграмма размещения.

***Разработка пользовательских интерфейсов. Оценка качества программного обеспечения.***

Разработка пользовательского интерфейса. Граф диалога с пользователем. Оценка качества программного обеспечения.

* + - * 1. Лабораторные занятия, их содержание

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | № раздела дисциплин  ы | Тема занятия | Форма проведения | Формируемые компетенции |
|  | 1 | Основы программирования (Линейные программы, Циклические программы, Массивы) | Лабораторная | ОПК-4, ОПК-5 |
|  | 1 | Оценка характеристик разработанных программ с помощью метрик Холстеда и метрик Джилба, оценка надежности программных средств с помощью модели Джелински-Моранды, эвристической модели и модели Нельсона | Лабораторная | ОПК-4, ОПК-5 |
|  | 2 | Работа со строками | Лабораторная | ОПК-4, ОПК-5 |
|  | 2 | Оценка степени отлаженности разрабатываемых приложений с помощью модели Миллса | Лабораторная | ОПК-4, ОПК-5 |
|  | 3 | Структурное тестирование разработанных приложений | Лабораторная | ОПК-4, ОПК-5 |
|  | 3 | Функциональное тестирование разработанных приложений | Лабораторная | ОПК-4, ОПК-5 |
|  | 4 | Работа с классами. Наследование | Лабораторная | ОПК-4, ОПК-5 |
|  | 4 | Разработка UML-диаграмм | Лабораторная | ОПК-4, ОПК-5 |
|  | 5 | Разработка многооконного приложения для работы с файлами | Лабораторная | ОПК-4, ОПК-5 |
|  | 5 | Разработка графа диалога пользователя | Лабораторная | ОПК-4, ОПК-5 |
|  | 5 | Оценка качества разработанного приложения | Лабораторная | ОПК-4, ОПК-5 |

* + - 1. **Учебно-методическое обеспечение самостоятельной работы студента и оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины**
  1. **Текущий контроль**

Текущий контроль производится путем защиты лабораторных работ

В конце каждого семестра проводится тестирование по изученным тема дисциплины. Сами тесты представлены в Системе управления курсами РГГМУ (http://moodle.rshu.ru)/

Критерии оценивания лабораторных работ.

* оценка «зачтено»: работа полностью выполнена. Даны полные ответы на вопросы по теме работы;
* оценка «не зачтено»: работа не выполнена или при защите студент не может ясно и четко ответить на поставленные вопросы.
  1. Методические указания по организации самостоятельной работы

Самостоятельная работа связана с изучением и конспектированием отдельных вопросов лекционного материала, выделенного преподавателем. Для успешного выполнения самостоятельной работы необходимо:

* + - в соответствии с заданной темой проработать соответствующий лекционный материал;
    - прочитать литературу из рекомендованного списка;
    - при необходимости осуществить поиск нужной информации в сети. Контроль выполнения самостоятельной работы обучающегося осуществляется собеседованием по определению понимания изученного материала.

***Контроль исполнения*** самостоятельных работ осуществляется преподавателем с участием студентов в форме обсуждения выполненных заданий.

* 1. Промежуточный контроль: экзамен (4 семестр)

Перечень вопросов для промежуточной аттестации – экзамен:

1. Технология программирования. Основные понятия. Этапы развития технологии программирования.

2. Жизненный цикл и этапы разработки программного обеспечения

3. Модели жизненного цикла программного обеспечения

4. Понятия эффективности и технологичности программного обеспечения. Модули и их свойства (сцепление и связность модулей)

5. Структурное программирование

6. Средства описания структурных алгоритмов (псевдокоды, схемы алгоритмов)

7. Средства описания структурных алгоритмов (Flow-формы, диаграммы Насси-Шнейдермана)

8. Правила оформления программ.

9. Разработка технического задания

10. Классификация моделей разрабатываемого программного обеспечения

11. Структурный подход. Диаграммы переходов состояний

12. Структурный подход. Функциональные диаграммы

13. Структурный подход. Диаграммы потоков данных

14. Структурный подход. Структуры данных и диаграммы отношений компонентов данных

15. Структурный подход. Сетевая модель данных (Диаграммы «сущность-связь»)

16. Проектирование программного обеспечения при структурном подходе. Структурная и функциональная схемы

17. Структурный подход. Структурные карты Константайна

18. Проектирование структур данных. Методика Джексона

19. UML- стандартный язык описания разработки прО1раммных продуктов с использованием объектного подхода

20. Диаграммы вариантов использования

21. Диаграмма классов. Отношения между классами

22. Диаграмма последовательностей

23. Диаграммы деятельности

24. Диаграмма пакетов

25. Диаграммы состояний объекта

26. Диаграмма кооперации

27. Диаграмма компонентов

28. Диаграмма размещения

29. Структурное тестирование. Тестирование базового пути

30. Структурное тестирование. Тестирование условий

31. Структурное тестирование. Тестирование циклов

32. Структурное тестирование. Тестирование потоков данных

33. Функциональное тестирование. Разбиение на классы эквивалентности и анализ граничных значений

34. Функциональное тестирование. Анализ причинно-следственных связей

35. Классификация ошибок

36. Методы отладки программного обеспечения

37. Разработка пользовательского интерфейса. Классификация диалогов и общие принципы их работы

38. Разработка пользовательского интерфейса. Граф диалога с пользователем

39. Оценка качества программного обеспечения

Образец билета:

Российский государственный гидрометеорологический университет Кафедра информационных технологий и систем безопасности ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1

Дисциплина «Методы программирования»

1. Классификация ошибок
2. Жизненный цикл и этапы разработки программного обеспечения
3. Напишите программу вычисляющую факториал введенного числа.

«Утверждаю»

Зав. кафедрой / /

Критерии выставления оценки

Оценка «**отлично**» ставится студенту, ответ которого содержит:

* + глубокое знание программного материала, а также основного содержания и новаций лекционного курса но сравнению с учебной литературой;
  + знание концептуально-понятийного аппарата всего курса; а также свидетельствует о способности:
  + самостоятельно критически оценивать основные положения курса;
  + увязывать теорию с практикой.

Оценка «отлично» не ставится в случаях систематических пропусков студентом семинарских и лекционных занятий по неуважительным причинам, а также неправильных ответов на дополнительные вопросы преподавателя.

Оценка «**хорошо**» ставится студенту, ответ которого свидетельствует о полном знании материала по программе, а также содержит в целом правильное, но не всегда точное и аргументированное изложение материала.

Оценка «хорошо» не ставится в случаях пропусков студентом семинарских и лекционных занятий по неуважительным причинам.

Оценка «**удовлетворительно**» ставится студенту, ответ которого содержит:

* + поверхностные знания важнейших разделов программы и содержания лекционного курса;
  + затруднения с использованием научно-понятийного аппарата и терминологии курса;
  + стремление логически четко построить ответ, а также свидетельствует о возможности последующего обучения.

Оценка «**неудовлетворительно**» ставится студенту, имеющему существенные пробелы в знании основного материала по программе, а также допустившему принципиальные ошибки при изложении материала.

**Примерные темы Курсовых Работ:**

Разработать криптографическую систему для следующих криптоалгоритмов:

1 Аффинная система подстановок Цезаря

2 Систем Цезаря с ключевым словом

3 Шифрующие таблицы Трисемуса

4 Биграммный шифр Плейфейра

5 Шифр Гронсфельда

6 Магические квадраты

7 Система шифрования Вижинера

8 Диск Альберти

9 Двойной квадрат Уитстона

10 Программная реализация роторной машины

11 Шифрующие таблицы с двойной перестановкой

12 Поворотная решетка Кардано

13 Шифр Ришелье

1. Учебно-методическое и информационное обеспечение дисциплины а) основная литература:
   1. Кувшинов, Д. Р. Основы программирования : учебное пособие для вузов / Д. Р. Кувшинов. — М. : Издательство Юрайт, 2018. — 104 с. — (Серия : Университеты России). — ISBN 978-5-534-07559-5. — Режим доступа : [www.biblio-online.ru/book/3D35AAB8-863B-47A8-BA39-ABF5D579204D](http://www.biblio-online.ru/book/3D35AAB8-863B-47A8-BA39-ABF5D579204D)
   2. Малявко, А. А. Формальные языки и компиляторы : учебное пособие для вузов / А. А. Малявко. — М. : Издательство Юрайт, 2018. — 429 с. — (Серия : Университеты России). — ISBN 978-5-534-04288-7. — Режим доступа : [www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3.](http://www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3)
   3. Огнева, М. В. Программирование на языке с++: практический курс : учебное пособие для бакалавриата и специалитета / М. В. Огнева, Е. В. Кудрина. — М. : Издательство Юрайт, 2018. — 335 с. — (Серия : Бакалавр и специалист). — ISBN 978-5-534-05123-0. — Режим доступа : www.biblio- online.ru/book/7670D7EC-AC37-4675-8EAE-DD671BC6D0E4
   4. Малявко, А. А. Формальные языки и компиляторы : учебное пособие для вузов / А. А. Малявко. — М. : Издательство Юрайт, 2018. — 429 с. — (Серия : Университеты России). — ISBN 978-5-534-04288-7. — Режим доступа : [www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3](http://www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3)
   5. Зыков, С. В. Программирование : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2018.

— 320 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-02444-

1. — Режим доступа : [www.biblio-online.ru/book/122D27F3-13E4-4095-8946-](http://www.biblio-online.ru/book/122D27F3-13E4-4095-8946-)

C619F0FCC5C3

* 1. A Byte of Python (Russian) Версия 2.02 / Swaroop C H (Перевод: Владимир Смоляр) Режим доступа: https://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.02.pdf

б) дополнительная литература:

* 1. Шишкин А.Д., Чернецова Е.А. Практикум "Программирование на языке Си" [Текст] : учеб. пособие / РГГМУ. - Санкт-Петербург : РГГМУ, 2003. - 51 с. - 25.00 р.
  2. Шишкин А.Д., Чернецова Е.А. Практикум по дисциплине "Языки программирования". Раздел: Программирование С и С++ [Текст] : практикум / РГГМУ. - 2-е изд., испр. и доп. - Санкт-Петербург : РГГМУ, 2012. - 68 с. - 29.43 р.
  3. Шишкин А.Д. Программирование на языке Си [Текст] : учебное пособие / РГГМУ. - Санкт-Петербург : РГГМУ, 2003. - 103 с. - 70.00 р.
  4. Назаров, С. В. Программирование в пакетах MS Office [Электронный ресурс] : учеб. пособие / С. В. Назаров, П. П. Мельников, Л. П. Смольников и др.; под ред. С. В. Назарова. - М.: Финансы и статистика, 2007. - 656 c.: ил. - ISBN 978-5-279-02926-6 - Режим доступа:<http://znanium.com/bookread2.php?book=369386>

в) программное обеспечение и Интернет-ресурсы:

*Программное обеспечение:*

* + - windows
    - office 2
    - Dev-C++ GNU General Public License
    - PyCharm: IDE

*Интернет-ресурсы*

* + - https://inf1.info/ - Планета информатики
    - https://www.onlinegdb.com/online\_python\_compiler
    - <http://taskcode.ru/>- Решение задач по программированию
    - [http://moodle.rshu.ru](http://moodle.rshu.ru/) - система управления курсами РГГМУ

*Информационно-справочные системы:*

* + - [https://biblio-online.ru](https://biblio-online.ru/) – ЭБС Юрайт
    - [http://znanium.com](http://znanium.com/) – ЭБС Знаниум
    - [http://www.prospektnauki.ru](http://www.prospektnauki.ru/) – ЭБС Проспект науки
    - [http://elib.rshu.ru](http://elib.rshu.ru/) ЭБС ГидроМетеоОнлайн
    - https://нэб.рф - Национальная электронная библиотека

*Профессиональные базы данных*

* + - Профессиональные базы данных не используются

1. Методические указания для обучающихся по освоению дисциплины

**(модуля)**

|  |  |
| --- | --- |
| Виды учебных занятий | Организация деятельности студента |
| Лекция | Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале,  необходимо сформулировать вопрос и задать преподавателю на  консультации или на лабораторном занятии. |
| Лабораторные | На лабораторных занятиях выполняются работы по программированию физических и математических задач, изученные во время лекций. Как правило, на каждом занятии студент должен показать результаты выполнения лабораторной преподавателю. После выполнения лабораторной работы студент готовится к ее защите и  защищает ее. |
| Внеаудиторная работа | представляет собой вид занятий, которые каждый студент организует и  планирует самостоятельно. Самостоятельная работа студентов включает самостоятельное изучение разделов дисциплины. |
| Подготовка к экзамену/ зачёту/  курсовой работе | При подготовке к экзамену, зачету и защите курсовой работы необходимо ориентироваться на конспекты лекций, рекомендуемую  литературу и выполненные лабораторные работы. |

1. **Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

|  |  |  |
| --- | --- | --- |
| Тема (раздел) дисциплины | Образовательные и информационные технологии | Перечень программного обеспечения и информационных  справочных систем |
| Методы программирования | Лекции. Лабораторные работы | [https://biblio-online.ru](https://biblio-online.ru/) [http://znanium.com](http://znanium.com/) [http://www.prospektnauki.ru](http://www.prospektnauki.ru/) [http://elib.rshu.ru](http://elib.rshu.ru/) https://нэб.рф  https://www.onlinegdb.com  windows  office  Dev-C++ GNU  PyCharm: IDE |

1. **Особенности освоения дисциплины для инвалидов и лиц с ограниченными возможностями здоровья**

Обучение обучающихся с ограниченными возможностями здоровья при необходимости осуществляется на основе адаптированной рабочей программы с использованием специальных методов обучения и дидактических материалов, составленных с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (обучающегося).

При определении формы проведения занятий с обучающимся-инвалидом учитываются рекомендации, содержащиеся в индивидуальной программе реабилитации инвалида, относительно рекомендованных условий и видов труда. При необходимости для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья создаются специальные рабочие места с учетом нарушенных функций и ограничений жизнедеятельности.

1. **Материально-техническое обеспечение дисциплины**

**Учебная аудитории для проведения занятий лекционного типа** – укомплектована специализированной (учебной) мебелью, набором демонстрационного оборудования и учебно-наглядными пособиями, обеспечивающими тематические иллюстрации, соответствующие рабочим учебным программам дисциплин (модулей).

Учебная аудитории для проведения занятий лабораторного типа -

компьютерный класс с ЛВС, связанной Интернетом

**Учебная аудитория для групповых и индивидуальных консультаций** - укомплектована специализированной (учебной) мебелью, техническими средствами обучения, служащими для представления учебной информации.

**Учебная аудитория для текущего контроля и промежуточной аттестации** - укомплектована специализированной (учебной) мебелью, техническими средствами обучения, служащими для представления учебной информации.

**Помещение для самостоятельной работы** – укомплектовано специализированной (учебной) мебелью, оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечено доступом в электронную информационно-образовательную среду организации

**Помещение для хранения и профилактического обслуживания учебного оборудования** – укомплектовано специализированной мебелью для хранения оборудования и техническими средствами для его обслуживания.