

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Райхерт Татьяна Николаевна
Должность: Директор
Дата подписания: 05.03.2023 11:20:03
Уникальный программный ключ:
c914df807d771447164c08ee17f8e2f93dde816b

Министерство просвещения Российской Федерации
Нижнетагильский государственный социально-педагогический институт (филиал)
федерального государственного автономного образовательного учреждения
высшего образования
«Российский государственный профессионально-педагогический университет»

Факультет естествознания, математики и информатики
Кафедра информационных технологий



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ Б1.В.01.01 ВЫСОКОУРОВНЕВЫЕ МЕТОДЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

Уровень высшего образования
Направление подготовки

Бакалавриат
09.03.03 Прикладная информатика

Профиль

«Прикладная информатика в управлении
ИТ-проектами»

Формы обучения

Очная, заочная

Рабочая программа дисциплины «Высокоуровневые методы информатики и программирования». Нижний Тагил: Нижнетагильский государственный социально-педагогический институт (филиал) ФГАОУ ВО «Российский государственный профессионально-педагогический университет», 2020. – 13 с.

Настоящая программа составлена в соответствии с требованиями федеральных государственных образовательных стандартов высшего образования по направлению 09.03.03 Прикладная информатика, профиль «Прикладная информатика в управлении IT-проектами».

Автор: кандидат пед. наук, доцент, доцент кафедры ИТ Н.В. Бужинская

Рецензент: зам.директора по ИТ
ИТ МУП НТТС Д.В. Виноградов

Программа рассмотрена и одобрена на заседании кафедры информационных технологий 9 апреля 2020 г., протокол № 9.

Заведующая кафедрой М. В. Мащенко

Программа рекомендована к печати методической комиссией факультета естествознания, математики и информатики 30 апреля 2020 г., протокол №8.

Председатель МК ФЕМИ Н. З. Касимова

Программа рассмотрена и утверждена на заседании Ученого совета факультета естествознания, математики и информатики 30 апреля 2020 г., протокол №8.

Декан ФЕМИ Т. В. Жуйкова

Главный специалист ОИР О. В. Левинских

© Нижнетагильский государственный социально-педагогический институт (филиал) ФГАОУ ВО «Российский государственный профессионально-педагогический университет», 2020.
© Бужинская Надежда Владимировна, 2020.

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы	4
3. Результаты освоения дисциплины	4
4. Структура и содержание дисциплины.....	5
4.1. Объем дисциплины, виды контактной и самостоятельной работы.....	5
4.2. Содержание и тематическое планирование дисциплины.....	6
4.3. Содержание тем дисциплины.....	8
5. Образовательные технологии.....	9
6. Учебно-методические материалы	9
6.1. Организация самостоятельной работы студентов.....	9
6.2. Организация текущего контроля и промежуточной аттестации	10
7. Учебно-методическое и информационное обеспечение	12
8. Материально-техническое обеспечение дисциплины	13

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины – отработка технологии применения основ программирования для решения задач будущей профессиональной деятельности.

Задачи:

1. Ознакомить студентов с современными технологиями программирования: объектно-ориентированным программированием, модульным программированием, компонентно-ориентированным программированием.
2. Ознакомить студентов с жизненным циклом программного продукта, начиная от этапа разработки и заканчивая его адаптацией соответственно требованиям заказчика.
3. Рассмотреть основные методы решения задач по программированию.
4. Продемонстрировать студентам способы тестирования программных продуктов.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Высокоуровневые методы информатики и программирования» является дисциплиной профильного модуля по направлению 09.03.03 Прикладная информатика. Реализуется кафедрой информационных технологий в 3-4 семестрах.

Данная дисциплина является продолжением изучения таких дисциплин как «Алгоритмизация и программирование», «Операционные системы», «Базы данных» и др. Полученный при изучении опыт деятельности может быть полезен студентам в выполнении учебных проектов и оформлении выпускных квалификационных работ.

3. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина направлена на формирование следующих профессиональных компетенций **ОПК-7, ПК-2, ПК-9.**

Код и наименование профессиональной компетенции		Код и наименование индикатора достижения профессиональной компетенции
ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения		ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. ОПК-7.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ОПК-7.3. Использует при решении профессиональных задач навыки программирования, отладки и тестирования прототипов программно-технических комплексов.
ПК-2.	Способность разрабатывать и адаптировать прикладное программное обеспечение.	ПК-2.1. Знает структуру и технологии разработки прикладного ПО. ПК-2.2. Знает современные языки и среды программирования. ПК-2.3. Умеет использовать основные технологии разработки программных продукты. ПК-2.4. Адаптирует прикладное программное обеспечение под нужды организации.

ПК-9. Способность проводить тестирование компонентов программного обеспечения ИС.	ПК-9.1. Знает основные подходы к тестированию ПО. ПК-9.2. Умеет составлять различные виды тестов для компонентов программного обеспечения ИС. ПК-9.3. Проводит тестирование ПО, правильно фиксирует результаты. ПК-9.4. Исправляет выявленные в ходе тестирования ошибки.
---	--

Таким образом, обучающийся после освоения дисциплины будет **знать:**

- современные подходы к программированию приложений, существующие парадигмы и технологии программирования, а также перспективы их развития;
- способы решения задач профессиональной деятельности на основе информационной и библиографической культуры;
- требования к тестированию прикладного программного обеспечения;
- требования информационной безопасности;
- структуру и технологии разработки ПО;
- основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий

Уметь:

- разрабатывать, тестировать, внедрять и адаптировать прикладное программное обеспечение;
- применять требования информационной безопасности при решении задач по программированию;
- проектировать и разрабатывать программные продукты с использованием современных технологий программирования;
- составлять различные виды тестов для компонентов программного обеспечения ИС;
- умеет проводить тестирование и исправлять выявленные в ходе тестирования ошибки.

Владеть:

- технологией программирования;
- способами разработки приложения, тестирования, внедрением и адаптации.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Объем дисциплины, виды контактной и самостоятельной работы

Общая трудоемкость дисциплины составляет 6 зач. ед. (216 часов), их распределение по видам работ представлено в таблице.

Распределение трудоемкости дисциплины по видам работ

Вид работы	Кол-во часов дневное	Кол-во часов заочное
Общая трудоемкость дисциплины по учебному плану	216	216
Контактная работа, в том числе:	74	20
Лекции	26	8
Лабораторные работы	48	12
Самостоятельная работа, в том числе:	142	196
Самоподготовка к текущему контролю знаний	106	187
Подготовка к экзамену	36	9

4.2. Тематический план

4.2.1. Тематический план для очной формы обучения

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час		Самостоятельная работа	Формы текущего контроля успеваемости
		Лекции	Лаб. работы		
Тема 1. Современные направления развития программирования	22	2		20	Проверка отчетов к лабораторным работам
Тема 2. Объектно-ориентированный подход в программировании	50	6	16	28	Проверка отчетов к лабораторным работам
Итого за 3 семестр	72	8	16	48	
Тема 3. Разработка приложений с графическим интерфейсом	46	10	16	20	Проверка отчетов к лабораторным работам
Тема 4. Жизненный цикл программных продуктов	30	4	6	20	Проверка отчетов к лабораторным работам
Тема 5. Применение высокоуровневых языков программирования для решения различных задач	32	4	10	18	Проверка отчетов к лабораторным работам
Экзамен	36			36	
Итого за 4 семестр	144	18	32	94	
Итого:	216	26	48	142	

4.2.1. Тематический план для заочной формы обучения

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час		Самостоятельная работа	Формы текущего контроля успеваемости
		Лекции	Лаб. работы		
Тема 1. Современные направления развития программирования	44	2		40	Проверка отчетов к лабораторным работам
Тема 2. Объектно-ориентированный подход в программировании	64	2	6	58	Проверка отчетов к лабораторным работам
Итого за 3 семестр	108	4	6	98	
Тема 3. Разработка приложений с графическим интерфейсом	36	2	4	30	Проверка отчетов к лабораторным работам
Тема 4. Жизненный цикл программных продуктов	34	2	2	30	Проверка отчетов к лабораторным работам
Тема 5. Применение высокоуровневых языков программирования для решения различных задач	29			29	Проверка отчетов к лабораторным работам
Экзамен	9			9	

Наименование разделов и тем дисциплины (модуля)	Всего, часов	Вид контактной работы, час		Самостоятельная работа	Формы текущего контроля успеваемости
		Лекции	Лаб. работы		
Итого за 4 семестр	108	4	6	98	
Итого:	216	8	12	196	

4.3. Практические занятия

№ п.п.	Наименование практических занятий	Кол-во ауд. часов
3 семестр		
1	Структуры данных	2
2	Классы	2
3	Методы	2
4	Решение задач	2
5	Реализация свойств ООП	2
6	Решение задач	2
7	Делегаты и индикаторы	2
8	Решение задач	2
	Итого:	16
4 семестр		
1	Интерфейс интегрированной среды разработки	2
2	Компиляция кода. Средства отладки. Настройка среды. Управление исходным кодом	2
3	Разработка форм	2
4	Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки	2
5	Обработка событий	2
6	Решение задач	2
7	Назначение технологии ADO.	2
8	Использование технологии ADO для организации доступа к данным	2
9	Основные подходы к тестированию	2
10	Правила написания тестов	2
11	Тестирование приложения. Составление документации	2
12	Использование объектно-ориентированного языка для разработки программных продуктов для автоматизации деятельности работников различных сфер	2
13	Выбор темы проекта. Постановка задачи. Обоснование выбора средств	2
14	Разработка приложения	2
15	Разработка приложения	2
16	Защита проекта	2
	Итого:	32

4.5. Содержание дисциплины

Тема 1. Современные направления развития программирования.

Парадигмы программирования.

Понятие парадигмы и технологии программирования. Сравнительная характеристика декларативного и императивного программирования; логической, функциональной, структурной, объектно-ориентированной и параллельной парадигм программирования: достоинства и недостатки, область применения.

Технологии программирования (объектно-ориентированное, модульное, компонентно-ориентированное) и область их применения.

Понятие «технология программирования». Новые направления развития программирования.

Тема 2. Объектно-ориентированный подход в программировании.

Реализация объектно-ориентированного подхода на одном из современных высокоуровневых языков программирования.

Сущность объектно-ориентированного подхода. Объектный тип данных. Переменные объектного типа.

Классы и объекты. Классификация объектов. Ограничение доступа к полям класса с помощью свойств.

Методы и события. События в среде Windows. Объявление класса. Создание экземпляра класса. Описание классов. Пространства имен. Конструкторы и деструкторы. Виды конструкторов. Свойства классов: наследование, инкапсуляция, полиморфизм. Перегрузка и переопределение методов на основании наследования.

Работа с динамическими массивами. Класс Array. Указатели. Работа с памятью компьютера. Реализация динамических структур данных с помощью массивов и указателей. Стеки. Очереди. Деревья.

Работа с XML-документами.

Тема 3. Разработка приложений с графическим интерфейсом.

Разработка пользовательского интерфейса. Виды интерфейса.

Понятие интерфейса. Особенности пользовательского интерфейса. Виды интерфейсов (командный, графический, звуковой) и способы их создания (MDI-технология, SDI-технология). Элементы интерфейса: окно, форма, меню, панель (лента). Использование библиотеки графических объектов для создания интерфейса. Создание MDI и SDI интерфейсов: разработка меню, панели инструментов, стандартных диалоговых окон, строки состояния, контекстного меню. Технология WPF и ее использование для создания пользовательских форм.

Интерфейс интегрированной среды разработки Visual Studio.

Компиляция кода. Средства отладки. Настройка среды. Управление исходным кодом.

Разработка форм. Класс Form. Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки. Процедуры обработки событий. Функция MsgBox и InputBox.

Обработка событий.

Обработка событий клавиатуры. Обработка событий мыши. Разработка справочной системы.

Тема 4. Жизненный цикл программных продуктов.

Создание и тестирование прикладных приложений на одном из современных высокоуровневых языков программирования.

Понятие тестирования. Виды тестирования. Подходы к тестированию. Сервисы для тестирования программных продуктов.

Модели разработки программных продуктов.

Жизненный цикл программного продукта. Способы описания жизненного цикла программного продукта. Оформление документации. Техническое задание. Руководство пользователя.

Тема 5. Применения высокоуровневых языков программирования для решения различных задач.

Искусственный интеллект

Понятие искусственного интеллекта. Особенности языка Пролог. Правила сопоставления термов в системе Пролог.

Парадигма визуального программирования. Визуальное и объектно-ориентированное программирование. Среды визуального программирования. Объектно-ориентированное программирование. Основы. Инкапсуляция, наследование, полиморфизм. Перегрузка операторов. Декораторы. Требования к проекту. Проектная деятельность по программированию.

Мобильные приложения

Обзор языков для создания мобильных приложений

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Теоретическая часть курса посвящена изучению возможностей программирования. Для ее изучения используются интерактивные лекции (проблемные, демонстрационные и др.).

Основными методами, используемыми на практических занятиях, будут: метод демонстрационных примеров, мастер-класс, практикум с использованием практико-ориентированных задач и проектная технология.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

6.1. Организация самостоятельной работы студентов

Темы занятий	Количество часов			Содержание самостоятельной работы	Формы контроля СРС
	Всего	Ауди-торн.	Сам. работа		
Тема 1. Современные направления развития программирования	22	2	20	Выполнение лабораторных работ	Проверка отчетов к лабораторным работам
Тема 2. Объектно-ориентированный подход в программировании	50	22	28	Выполнение лабораторных работ	Проверка отчетов к лабораторным работам
Итого за 3 семестр	72	24	48		
Тема 3. Разработка приложений с графическим интерфейсом	46	26	20	Составление глоссария	Проверка правильности составления глоссария
Тема 4. Жизненный цикл программных продуктов	30	10	20	Выполнение лабораторных работ	Проверка отчетов к лабораторным работам
Тема 5. Применение высокоуровневых языков программирования для решения различных	32	14	18	Выполнение лабораторных работ	Проверка отчетов к лабораторным работам

Темы занятий	Количество часов			Содержание самостоятельной работы	Формы контроля СРС
	Всего	Ауди-торн.	Сам. работа		
задач					
Экзамен	36	0	36		
Итого за 4 семестр	144	50	94		
Итого:	216	74	142		

В рамках самостоятельной работы студентов предусмотрено выполнение мини-проектов, которые рассчитаны на самостоятельную внеаудиторную работу студентов. Цель заданий – закрепление знаний, полученных на лабораторных занятиях. Задания можно выполнять дома или в компьютерном классе вуза. В процессе работы можно пользоваться методическими указаниями для выполнения лабораторных работ.

Проект 1. Создание банковского калькулятора для пересчета валют, расчета по кредитам и вкладам

Студент должен показать владение основными конструкциями программирования. В проекте должно быть предусмотрен расчет кредита методом простых и сложных процентов.

Проект 2. Создание индивидуального мини-проекта, представляющего собой решение практико-ориентированной задачи

Тему проекта студент формулирует сам, например «Разработка виртуальной экскурсии по г. Нижний Тагил», «Разработка базы данных по результативности игроков чемпионата мира по футболу». В процессе разработки студент должен показать владение технологиями программирования.

Примерная тематика индивидуальных проектов

1. Разработка электронного календаря-еженедельника, позволяющего планировать текущие дела и напоминать о них согласно текущей дате.
2. Разработка электронного приложения «Склад», позволяющего заносить записи о товарах, их ценах и количестве, сортировать, находить нужный товар, иллюстрировать текущее положение в виде диаграммы и удалять купленные товары.
3. Разработка налогового калькулятора индивидуального предпринимателя.
4. Разработка валютного калькулятора с возможностью добавления новых валют и постоянного обновления курсов, а также хранением курсов в течение недели и иллюстрации их изменения в виде графика.
5. Разработка туристического калькулятора для расчета стоимости путевки в зависимости от входных параметров.
6. Разработка текстового редактора.
7. Разработка графического редактора.
8. Разработка мультимедиа плеера, поддерживающего различные форматы файлов.
9. Разработка математического редактора для работы с квадратными матрицами.
10. Разработка стратегической игры для двух игроков.

Проект 3. Разработка игры для детей младшего школьного возраста

Проект подразумевает разработку игры для детей младшего школьного возраста, иллюстрирующей события мыши и клавиатуры. При разработке игры необходимо учесть особенности детей данного возраста.

Критерии оценки проектов

- соответствие предложенной тематике;
- обоснованность структуры и выбранных форм;
- полнота;
- результативность;
- оригинальность;
- оптимальность используемых алгоритмов;

- оптимальность использования базовых элементов;
- удобство интерфейса пользователя.
- наглядность представления проекта;
- убедительность при обосновании и защите проекта;
- уровень владения материалом;
- умение вести дискуссию.

2 – компонент присутствует в полном объеме, 1- компонент присутствует, но есть ошибки, 0- компонент отсутствует. Оценка рассчитывается соответственно %, представленным в положении о НБРС НТСГИ (ф) РГППУ.

Общее количество баллов – 24.

«Отлично» – > 21 балла.

«Хорошо» – 17 – 20 баллов.

«Удовлетворительно» – 12 – 16 баллов.

«Не удовлетворительно» – < 12 баллов.

6.2. Организация текущего контроля и промежуточной аттестации

Текущий контроль усвоения знаний ведется по итогам представления выполненных самостоятельных заданий и защиты отчетов по лабораторным работам; участия в дискуссиях на лекционных занятиях, проверки составленного глоссария и результатов тестирования. Текущий контроль учебных достижений студентов может быть проведен с использованием накопительной балльно-рейтинговой системы оценки в соответствии с Положением о НБРС.

Промежуточная аттестация по данной дисциплине проводится в форме экзамена в четвертом семестре, на которых теоретические знания студентов проверяются в ходе устного ответа на вопросы, а практические – в процессе решения практической задачи.

ПРИМЕРНЫЕ ВОПРОСЫ К ЭКЗАМЕНУ

1. Разработка форм. Класс Form. Базовые элементы управления: метка, текстовое поле, кнопка, флажок, переключатель, списки.
2. Назначение технологии ADO. Использование технологии ADO для организации доступа к данным.
3. Сущность объектно-ориентированного подхода. Объектный тип данных. Переменные объектного типа.
4. Классы и объекты. Классификация объектов. Ограничение доступа к полям класса с помощью свойств.
5. Методы и события.
6. Конструкторы и деструкторы. Виды конструкторов.
7. Свойства классов: наследование, инкапсуляция, полиморфизм.
8. Перегрузка и переопределение методов на основании наследования.
9. Реализация динамических структур данных с помощью массивов и указателей.
10. Стеки. Очереди. Деревья.
11. Работа с XML-документами.

ПРИМЕРНЫЕ ПРАКТИЧЕСКИЕ ЗАДАНИЯ

1. С помощью делегатов оформите вопрос теста. В качестве значения строковой переменной можно указать `string s=""`.
2. Напишите программу, будут меняться значения a и b. Если a положительное число – то его значения меняется на отрицательное, для b – наоборот. Добавьте событие с обработчиком, которое выводит на экран надпись, что пользователь ввел 0. Указание – для изменения значения a или b воспользуйтесь методом `set`, а для возврата значения – методом `get`.

3. Напишите методы для проверки, правильно ли введены стороны (они должны быть >0) и вывода значения этих сторон на экран.

4. Создайте класс для ввода длин сторон треугольника. Напишите конструктор для данного класса, создайте экземпляр класса и выведите информацию на экран.

Критерии оценки устного ответа на вопрос

- полнота ответа;
- лаконичность ответа и умение выделить главное;
- соответствие современным достижениям науки;
- логичность ответа и умение построить завершенную монологическую речь;
- научно-популярный (деловой) стиль изложения;
- наличие практических примеров из жизни или профессиональной деятельности.

Критерии оценки практического задания

- работоспособность алгоритмов;
- лаконичность решения и умение выделить главное;
- рациональность решения.

Критерии оценки

- Не удовлетворительно - низкий уровень знаний и умений по курсу (студент в деятельности совершает ошибки и не способен к их исправлению, не может ответить на теоретический вопрос, выполнить практическое задание).

- Удовлетворительно – достаточный уровень знаний и умений по курсу (деятельность студента не носит целенаправленный характер, однако отличается стремлением самостоятельно ставить перед собой цели; может определить основные понятия дисциплины);

- Хорошо – средний уровень знаний и умений по курсу (деятельность студента носит целенаправленный характер, отличается стремлением самостоятельно ставить перед собой различные цели: знает методологию предметной области);

- Отлично – высокий уровень знаний и умений (в деятельности студентов отмечается четкость, действенность и реальность целей самообразования; самообразование становится устойчивым, целенаправленным, систематичным).

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Основная литература

1. Бабушкина, И.А. Практикум по объектно-ориентированному программированию [Электронный ресурс] : учебное пособие. М.: Издательство «Лаборатория знаний», 2015. 369 с. URL: <https://e.lanbook.com/book/66121> (дата обращения: 10.06.2019).

2. Медведев М.А. Программирование на СИ# [Электронный ресурс]: учебное пособие/ Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2015. 64 с.: URL: <http://www.iprbookshop.ru/69667.html> (дата обращения: 12.07.2019).

3. Николаев Е.И. Объектно-ориентированное программирование [Электронный ресурс]: учебное пособие. Ставрополь: Северо-Кавказский федеральный университет, 2015. 225 с. Режим доступа: <http://www.iprbookshop.ru/62967.html> (дата обращения: 12.08.2019).

Дополнительная литература

1. Одинцов И. Профессиональное программирование. Системный подход, 2 изд. / И. Одинцов. - Санкт-Петербург : БХВ-Петербург, 2014. - 624 с. - ISBN 978-5-9775-1293-0. - URL: <https://ibooks.ru/reading.php?productid=18535> (дата обращения: 13.03.2020). - Текст: электронный.

2. Павловская Т.А. С/С++. Процедурное и объектно-ориентированное программирование. Учебник для вузов. Стандарт 3-го поколения / Т.А. Павловская. - Санкт-Петербург : Питер, 2015. - 496 с. - ISBN 978-5-496-00109-0. - URL: <https://ibooks.ru/reading.php?productid=341427> (дата обращения: 13.03.2020). - Текст: электронный.

Интернет-ресурсы

1. Академия ALT Linux. Практикум по алгоритмизации и программированию на Python. [Электронный ресурс]. URL: <https://www.intuit.ru/studies/courses/3489/731/info> (дата обращения: 18.02.2020).

2. Единое окно доступа к образовательным ресурсам: Федеральный портал. URL: <http://window.edu.ru/window/library>. (дата обращения: 09.02.2020). Режим доступа: свободный. Текст: электронный.

3. Объектное программирование в классах на С# 3.0. [Электронный ресурс]. URL: <https://www.intuit.ru/studies/courses/941/413/info> (дата обращения: 18.02.2020).

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебная аудитория 201Аа: 11 посадочных мест для студентов, рабочее место преподавателя, компьютеры – 12 шт., маркерная доска, робототехнические наборы Arduino, Lego MindStorms Ev3.

Пакет офисных программ: Office Standard 2016 Russian OLP NL Academic Edition.

Акт предоставления прав № IT021617 от 12.02.2016 г.

Microsoft Visio,

Microsoft OneNote,

Microsoft Project,

Microsoft SharePoint

Браузеры [Firefox](#), [Google Chrome](#), [Яндекс.Браузер](#)

Бесплатное ПО:

GIMP, Inkscape, Paint Net

7-Zip

Arduino IDE

Lego Digital Designer

Lego MindStorms Education Ev3

Python 3.6.