



Федеральное государственное бюджетное образовательное учреждение высшего образования
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

УТВЕРЖДАЮ

И.о. проректора

А.В. Троицкий

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ
И ТЕХНОЛОГИЙ**

КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

**РАБОЧАЯ ПРОГРАММА
ДИСЦИПЛИНЫ (МОДУЛЯ)**

«СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

Направление подготовки: 15.03.05 Конструкторско-технологическое обеспечение машиностроительных производств

Направленность (профиль): Технология машиностроения

Уровень высшего образования: бакалавриат

Форма обучения: очная, заочная

Королёв
2023


Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе основной профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.

Автор: Парафейников И.В. Рабочая программа дисциплины (модуля): «Современные языки программирования» – Королев МО: «Технологический университет», 2023.

Рецензент: к.т.н., с.н.с. Привалов В.И.

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавров 15.03.05 Конструкторско-технологическое обеспечение машиностроительных производств и Учебного плана, утвержденного Ученым советом Университета. Протокол № 9 от 11 апреля 2023 г.

Рабочая программа рассмотрена и одобрена на заседании кафедры:

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Баранова О.М. к.т.н. 	Баранова О.М. к.т.н.	Баранова О.М. к.т.н.	Баранова О.М. к.т.н.
Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания кафедры	№ 7 от 03.04.2023 г.			

Рабочая программа согласована:

Руководитель ОПОП  д.т.н., профессор Пашковский И.Э.

Рабочая программа рекомендована на заседании УМС:

Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания УМС	№ 5 от 11.04.2023 г.			

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Целью изучения дисциплины является:

1) формирование представления о современном подходе к решению различных задач прикладной области с помощью ЭВМ;

2) подготовка бакалавров к использованию современных компьютерных технологий и алгоритмов в будущей производственно-технической, проектно-конструкторской и исследовательской деятельности.

В процессе обучения студент приобретает и совершенствует следующие компетенции.

Общепрофессиональные компетенции:

ОПК-6 – 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.

Основными задачами дисциплины являются:

- ознакомление бакалавров по данному направлению обучения с основными инструментальными и программными средствами компьютерного решения задач;

- обучение технологии структурного и модульного программирования;

- приобретение бакалаврами навыков написания, отладки и тестирования, документирования надежных, безопасных и легко модифицируемых программ.

Показателями освоения компетенций служат следующие индикаторы:

Трудовые действия:

- осуществляет выбор информационных технологий для решения задач автоматизации, роботизации и проектирования технологических процессов машиностроительного производства;

- получает и обрабатывает информацию из различных источников, используя современные информационные технологии, критически осмысливает полученную информацию, на основе чего разрабатывает алгоритмы и компьютерные программы, пригодные для практического применения.

Необходимые умения:

- умеет выбирать современные информационные технологии и программные средства при решении задач машиностроительного производства;

- владеет навыками применения современных информационных технологий при решении задач профессиональной деятельности;

- умеет разрабатывать с использованием CALS-технологий на базе системного подхода последовательность решения поставленной задачи, определять алгоритмы и программные решения.

Необходимые знания:

- знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности;

- знает алгоритмы решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Современные языки программирования» относится к обязательной части блока 1 основной профессиональной образовательной программы по направлению подготовки 15.03.05 Конструкторско-технологическое обеспечение машиностроительных производств.

Изучение данной дисциплины базируется на отдельных разделах изученной ранее дисциплины «Информатика», и компетенциях: УК-1; ОПК-6, 10.

Знания и компетенции, полученные при освоении дисциплины, являются базовыми при изучении последующих дисциплин: «Программирование обработки деталей на станках с ЧПУ», «САПР технологических процессов» и выполнении выпускной квалификационной работы.

3. Объем дисциплины (модуля) и виды учебной работы

Общая трудоемкость дисциплины составляет 3 зачетных единицы, 108 часов.

Таблица 1

Виды занятий	Всего часов	Семестр 2	Семестр 3	Семестр 4	Семестр 5
Общая трудоемкость	108		108		
ОЧНАЯ ФОРМА ОБУЧЕНИЯ					
Аудиторные занятия	64		48		
Лекции (Л)	16		16		
Практические занятия (ПЗ)	48		16		
Лабораторные работы (ЛР)	-		26		
Практическая подготовка	-		-		
Самостоятельная работа	44		78		
Курсовые работы (проекты)	КР		-		
Расчетно-графические работы	РГР		-		
Контрольная работа, домашнее задание	кр		+		
Текущий контроль знаний	Тест		+		
Вид итогового контроля	Зачет / Экзамен		Зачет		
ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ					
Аудиторные занятия	16			16	
Лекции (Л)	8			8	
Практические занятия (ПЗ)	8			8	
Лабораторные работы (ЛР)	-			-	
Практическая подготовка	-			-	
Самостоятельная работа	92			92	
Курсовые работы (проекты)	КР			-	
Расчетно-графические работы	РГР			-	
Контрольная работа, домашнее задание	кр			+	
Вид итогового контроля	Зачет / Экзамен			Зачет	

4. Содержание дисциплины

4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час. очное/заочное	Прак. занятия, час. очное/заочное	Лаб. раб., час. очное/заочное	Занятия в интерактивной форме, час очное/заочное	Практическая подготовка, час очное/заочное	Код компетенций
Тема 1. Введение в курс «Современные языки программирования»	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 2. Понятие системы программирования	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 3. Классификация ЯП	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 4. Основные принципы программирования на языках высокого уровня	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 5. Технология структурного и модульного программирования	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 6. Стандартные библиотеки и их подпрограммы	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 7. Технология ООП	2/1	6/1	-	2/0,5	-	ОПК-6,10
Тема 8. Методы тестирования, отладки и сопровождения программ	2/1	6/1	-	2/0,5	-	ОПК-6,10
	16/8	48/8	-	16/4	-	

4.2. Содержание тем дисциплины

Тема 1. Введение в курс «Современные языки программирования».
Основные понятия. Алгоритмизация. Язык программирования (ЯП). Программное обеспечение (ПО).

Тема 2. Понятие системы программирования
Реализация языка программирования. Основные компоненты среды программирования. Программирование вычислений. Системное программирование. Прикладное программирование.

Тема 3. Классификация ЯП.

Многообразие современных языков программирования. Концепция языков программирования. Классификация языков программирования по степени детализации предписаний. Генеалогия языков программирования высокого уровня. Спецификация программ и стандартизация ЯП.

Тема 4. Основные принципы программирования на языках высокого уровня.

Архитектура и возможности семейства языков программирования высокого уровня. Представление основных алгоритмических структур. Этапы решения задач на ЭВМ: постановка задачи, анализ и исследование модели, разработка алгоритма.

Тема 5. Технология структурного и модульного программирования

Типы данных. Структуры данных. Представление основных управляющих структур программирования. Процедуры и функции. Модули.

Тема 6. Стандартные библиотеки и их подпрограммы

Стандартные модули систем программирования на ЯП С/С++ и Питон.

Тема 7. Технология ООП

Основные принципы объектно-ориентированного программирования. Поддержка ООП в современных ЯП С++ и Питон.

Тема 8. Методы тестирования, отладки и сопровождения программ

Этапы решения задач на ЭВМ: тестирование и отладка программ. Сопровождение программы. Критерии качества программы.

5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине

1. Методические указания для обучающихся по освоению дисциплины.
2. Методические указания для обучающихся по выполнению контрольных (или домашних) работ.

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Структура фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Современные языки программирования» приведена в Приложении 1.

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Бедердинова О.И. Программирование на языках высокого уровня: учеб. пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. – М.: ИНФРА-М, 2019. – 159 с. – Текст: электронный.
- URL: <https://znanium.com/catalog/product/1044396>
- Режим доступа: по подписке.
2. Основы алгоритмизации и программирования на Python: учеб. пособие / С.Р. Гуриков. – М.: ФОРУМ: ИНФРА-М, 2017. – 343 с. — (Высшее образование: Бакалавриат).
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=924699>
- Режим доступа: по подписке.
3. Программирование на языке высокого уровня. Программирование на языке C++: учеб. пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; под ред. Л.Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2018. – 512 с.
- URL: <http://znanium.com/catalog.php?bookinfo=918098>
- Режим доступа: по подписке.
4. Шелудько В.М. Язык программирования высокого уровня Python: функции, структуры данных, дополнительные модули / В.М. Шелудько; Министерство науки и высшего образования РФ, Южный федеральный университет, Институт компьютерных технологий и информационной безопасности. – Ростов-на-Дону; Таганрог: Южный федеральный университет, 2017. – 108 с.
- URL: <http://biblioclub.ru/index.php?page=book&id=500060>
- Режим доступа: по подписке.

Дополнительная литература:

1. Колдаев В.Д. Численные методы и программирование: учеб. пособие / В.Д. Колдаев; под ред. проф. Л.Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2017. – 336 с.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=672965>
- Режим доступа: по подписке.
2. Программирование графики на C++. Теория и примеры: учеб. пособие / В.И. Корнеев, Л.Г. Гагарина, М.В. Корнеева. — М.: ИД «ФОРУМ»: ИНФРА-М, 2017. — 517 с. + Доп. материалы [Электронный ресурс; Режим доступа <http://www.znanium.com>]. – (Высшее образование). – www.dx.doi.org/10.12737/23113.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=562914>
- Режим доступа: по подписке.
3. Основы алгоритмизации и программирования: учебное пособие / под ред. проф. Л. Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2017. – 416 с.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=902236>
- Режим доступа: по подписке.

4. Программирование на C++ с погружением: практические задания и примеры кода - М.: НИЦ ИНФРА-М, 2016. - 80 с.: 60x90 1/16
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=563294>
- Режим доступа: по подписке.

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://biblioclub.ru/> - Университетская библиотека ONLINE
2. <http://www.znanium.com/> - Электронно-библиотечная система
3. <http://www.en.edu.ru/> - естественнонаучный образовательный портал
4. <http://www.academyit.ru/> - академия АЙТИ.

9. Методические указания для обучающихся по освоению дисциплины

Методические указания для обучающихся по освоению дисциплины, приведены в Приложении 2.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения:

NotePad++, MSOffice, GCC + MinGW, Python 3, PyCharm.

Информационные справочные системы:

1. Ресурсы информационно-образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Современные языки программирования».

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия:

- аудитория, оснащенная презентационной техникой (проектор, экран), интерактивной доской Smart Board;
- Рабочее место преподавателя, оснащенное компьютером с доступом в Интернет.

Практические занятия:

- учебный класс, оснащенный вычислительной техникой (ПК), современные лицензионные программно-технические средства: операционная система не ниже Windows XP; NotePad++, MSOffice, GCC + MinGW, Python 3, PyCharm, рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ
«СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ»**

Направление подготовки: 15.03.05 Конструкторско-технологическое
обеспечение машиностроительных производств

Направленность (профиль): Технология машиностроения

Уровень высшего образования: бакалавриат

Форма обучения: очная, заочная

Королев
2023

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п./ п.	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся приобретает:		
				Трудовые действия	Необходимые умения	Необходимые знания
1.	ОПК-6	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	Тема 1-8.	Осуществляет выбор информационных технологий для решения задач автоматизации, роботизации и проектирования технологических процессов машиностроительного производства.	Умеет выбирать современные информационные технологии и программные средства при решении задач машиностроительного производства; Владеет навыками применения современных информационных технологий при решении задач профессиональной деятельности.	Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.
2.	ОПК-10	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения.	Тема 1-8.	Получает и обрабатывает информацию из различных источников, используя современные информационные технологии, критически осмысливает полученную информацию, на основе чего разрабатывает алгоритмы и компьютерные программы, пригодные для практического применения.	Умеет разрабатывать с использованием CALS-технологий на базе системного подхода последовательность решения поставленной задачи, определять алгоритмы и программные решения.	Знает алгоритмы решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Код компетенции	Инструменты, оценивающие сформированность компетенции	Показатель оценивания компетенции	Критерии оценивания компетенции на различных этапах формирования и шкалы оценивания
ОПК-6,10	Контрольная работа	<p>А) полностью сформирована (компетенция освоена на высоком уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> •компетенция освоена на продвинутом уровне – 4 балла; •компетенция освоена на базовом уровне – 3 балла; <p>В) не сформирована (компетенция не освоена) – 2 и менее баллов</p>	<p>Проводится устно с использованием мультимедийных систем, а также с использованием технических средств</p> <p>Время, отведенное на процедуру – 10 - 15 мин.</p> <p>Неявка – 0.</p> <p>Критерии оценки:</p> <ol style="list-style-type: none"> 1.Соответствие представленной презентации заявленной тематике (1 балл). 2.Качество источников и их количество при подготовке доклада и разработке презентации (1 балл). 3.Владение информацией и способность отвечать на вопросы аудитории (1 балл). 4.Качество самой представленной презентации (1 балл). 5.Оригинальность подхода и всестороннее раскрытие выбранной тематике (1 балл). <p>Максимальная сумма баллов - 5 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля. Оценка проставляется в электронный журнал.</p>

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Примерная тематика контрольных работ:

<p>Программа должна содержать меню и ввод-вывод в окна на экране. Необходимо смотреть контроль ошибок пользователя при вводе данных.</p> <p>При разработке программы следует применить технологию нисходящего проектирования, логически законченные фрагменты оформить в виде подпрограмм, все необходимые данные которым передаются через список параметров. Использование глобальных переменных следует избегать.</p>	
1.	<p>Имеется ведомость результатов экзаменационной сессии студенческой группы, которая хранится в файле на диске и для каждого студента содержит фамилию, инициалы и оценки по пяти предметам. Количество студентов в группе не превышает 20 человек. Написать программу, с помощью которой можно корректировать и дополнять список, и получать:</p> <ul style="list-style-type: none"> ▪ Список студентов; ▪ Список студентов, сдавших экзамены только на «5»; ▪ Список студентов, имеющих тройки; ▪ Список студентов, имеющих двойки. При этом студент, имеющий более чем одну двойку, исключается из списка.
2.	<p>Предприятие имеет местную телефонную станцию на 20 номеров. Телефонный справочник данного предприятия для каждого номера телефона содержит номер телефона и список служащих, сидящих в данном помещении.</p> <p>Создать программу, которая:</p> <ul style="list-style-type: none"> ▪ корректирует базу; ▪ по номеру телефона выдает номер помещения и список сидящих в нем людей; ▪ по номеру помещения выдает номер телефона; ▪ по фамилии выдает номер телефона и номер помещения; <p>Номер телефона – двузначный. В одном помещении может находиться от одного до четырех служащих.</p>
3.	<p>В гостинице имеется 15 номеров, из них 5 одноместных и 10 двухместных.</p> <p>Составить программу, которая заполняет и (или) корректирует данные о жильцах и по фамилии определяет номер, где проживает жилец. Программа запрашивает фамилию жильца.</p> <ul style="list-style-type: none"> ▪ Если жильца с такой фамилией нет, об этом выдается сообщение. ▪ Если жилец с такой фамилией в гостинице единственный, программа выдает фамилию жильца и номер проживания. ▪ Если в гостинице проживает два или более жильцов с такой фамилией, программа дополнительно запрашивает инициалы.
4.	<p>В текстовом файле хранится список служащих. Для каждого служащего указаны фамилия и инициалы, название занимаемой должности, год поступления на работу и оклад.</p> <p>Написать программу, выполняющую следующие действия:</p> <ul style="list-style-type: none"> ▪ корректировку или дополнение списка с клавиатуры; ▪ сортировку по фамилии, окладу или году поступления; ▪ вывод на экран информации о служащем, фамилия которого введена с клавиатуры; ▪ запись списка в файл под тем же или новым именем.

5.	<p>Расписание электричек хранится в текстовом файле. Каждая запись содержит название пункта назначения, пометки типа «СВ», «ПВ», «КСВ» и время отправления.</p> <p>Написать программу, выполняющую следующие действия:</p> <ul style="list-style-type: none"> ▪ корректировку или дополнение расписания с клавиатуры; ▪ сортировку по станции назначения или по времени отправления; ▪ вывод на экран информации о поездах, отходящих после введенного времени; ▪ запись расписания в файл под тем же или новым именем.
-----------	---

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Формой контроля знаний по дисциплине «Современные языки программирования» являются две текущие аттестации в виде тестов и одна итоговая в виде зачета.

Неделя текущего контроля	Вид оценочного средства	Код компетенций, оценивающий знания, умения, навыки	Содержание оценочного средства	Требования к выполнению	Срок сдачи (неделя семестра)	Критерии оценки по содержанию и качеству с указанием баллов
Согласно графика учебного процесса	1-е тестирование	ОПК-6,10	25 вопросов	Компьютерное тестирование; время, отведенное на процедуру - 30 минут,	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка – Удовлетворительно – от 51% правильных ответов. Хорошо - от 75%. Отлично – от 85%.
Согласно графика учебного процесса	2-е тестирование	ОПК-6,10	25 вопросов	Компьютерное тестирование; время, отведенное на процедуру – 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка – 0 Удовлетворительно – от 51% правильных ответов. Хорошо – от 75%.

						Отлично – от 85%. Максимальная оценка – 5 баллов.
Согласно графика учебного процесса	Зачет	ОПК-6,10	3 вопроса	Зачет проводится в письменной форме, путем ответа на вопросы. Время, отведенное на процедуру – 30 минут.	Результаты предоставляются в день проведения зачета	Критерии оценки: «Зачтено»: - знание основных понятий предмета; - умение использовать и применять полученные знания на практике; - работа на практических занятиях; - знание основных научных теорий, изучаемых предметов; - ответ на вопросы билета. «Не зачтено»: демонстрирует частичные знания по темам дисциплин; - незнание основных понятий предмета; - неумение использовать и применять полученные знания на практике; - не работал на практических занятиях; - не отвечает на вопросы.

4.1. Типовые вопросы, выносимые на тестирование

Тесты используются в режиме промежуточного контроля. По форме заданий выбраны закрытые тесты (с выборочным ответом). Каждому вопросу соответствует один вариант ответа.

История языков программирования высокого уровня. Структурное программирование.

1. Основное требование, предъявляемое к программе:

- надежность
- сложность
- простота
- компактность

2. Какое программирование называют «стиль спагетти»?

- программирование с бесконечными переходами
- программирование на языке низкого уровня
- программирование без «goto»
- программирование на языке высокого уровня

3. Создание любой программы начинается...

- с постановки задачи
- разработки внутренних структур данных
- со структурного программирования
- с проектирования

4. Создателем языка С является ...

- Никлаус Вирт
- Конрад Цузе
- Деннис Ритчи
- Джон Бэкус

5. Структурное программирование – это ...

- проверка исходных данных и соответствующих им наборов эталонных решений.
- процесс, посредством которого проверяется правильность программы.
- дисциплина, обучающая языку программирования высокого уровня.
- дисциплина, требующая, чтобы любая программа строилась из ограниченного набора типовых (базовых) управляющих структур и их композиций.

6. К базовым алгоритмическим структурам относятся:

- ветвление;
- блок;
- переход;
- цикл;
- линейный алгоритм (цепочка);

7. На этапе проектирования нужно стремиться спланировать программу таким образом, чтобы

- вносить изменения в нее было как можно сложнее
- вносить изменения в нее было как можно проще
- ее документация была как можно проще
- ее документация была как можно сложнее

8. Каков минимальный набор управляющих структур, с помощью которого можно представить любой алгоритм?

- 1
- 2
- 3
- 4

9. Согласно теореме структурирования любой алгоритм можно представить в виде двух управляющих структур. Каких?

- «цепочка»
- «ветвление»
- цикл с предусловием
- цикл с постусловием
- цикл с конечным числом итераций

10. Характерной особенностью базовой алгоритмической структуры «цепочка» является наличие в ней...

- одного входа и двух выходов;
- одного входа и одного выхода;
- двух входов и двух выходов;
- двух входов и одного выхода;

11. Главная цель, к которой должен стремиться программист, при решении поставленной задачи, - получить

- легко читаемую программу, возможно, более простой структуры
- универсальную программу
- более сложную программу, чем созданная ранее
- «экзотическую» программу

12. Языками программирования являются:

- Лого
- Турбо Алгол
- Пролог
- Перл
- Симула

13. Следуя правилам программирования, нужно

- использовать в программе числа в явном виде
- избегать использования в программе чисел в явном виде
- константы называть сложными именами
- назначать переменным имена, скрывающие их смысл

14. Какие компоненты образуют алгоритмический язык?

- семантика
- синтаксис
- служебные слова
- алфавит

15. В какой сфере язык Фортран нашёл наибольшее применение?

- в сфере научно-технических расчётов
- в торговой сфере
- в программировании удалённого доступа
- в сфере дизайна и анимации

16. Укажите основные преимущества алгоритмических языков по сравнению с машинными?

- Формат предложений достаточно громоздок и неудобен для использования.
- Алфавит алгоритмического языка значительно шире машинного.
- Набор операций, допустимых для использования, не зависит от набора машинных операций.
- Требуемые операции задаются с помощью общепринятых математических обозначений.
- Алфавит алгоритмического языка значительно уже машинного.

17. Первым языком, в котором появился достаточный набор управляющих операторов, позволяющих писать программу без «goto», является:

- Бейсик
- Алгол
- Паскаль
- Кобол
- Фортран

18. Данные делятся на:

- константы
- переменные
- операции
- операторы

19. Выражения алгоритмического языка предназначаются для выполнения необходимых вычислений и состоят из ...

- констант, переменных, указателей функций, объединённых знаками операций.
- констант и переменных.
- переменных и указателей функций.
- стандартных функций.

20. К критериям качества программы относятся:

- эффективность
- объем внешней памяти
- документированность
- время выполнения

21. Под проектированием программы понимается

- проверка исходных данных и соответствующих им наборов эталонных реакций
- процесс, посредством которого проверяется правильность программы
- создание технического задания
- определение общей структуры и взаимодействия модулей

22. Спецификация интерфейсов – это

- запись программы на языке программирования по готовому алгоритму
- представление алгоритмов
- определение способов взаимодействия подзадач
- программирование задач

23. Процесс проектирования задачи является

- линейным
- итерационным
- ветвящимся
- кольцевым

24. Представление алгоритма решения задачи в виде последовательности подзадач называется

- процедурной декомпозицией
- спецификацией интерфейсов
- структурным программированием
- нисходящим тестированием

25. При структурном программировании организуется кодирование программы по принципу:

- «сверху вниз»
- «снизу вверх»
- «слева направо»
- «справа налево»

4.2. Типовые вопросы, выносимые на зачет

1. Среды программирования. Реализация языка программирования.
2. Типы языков программирования. Классификация языков программирования в зависимости от степени детализации предписаний.
3. Генеалогия языков программирования высокого уровня.
4. Преимущества алгоритмических языков относительно машинных.
5. Транслятор: компилятор, интерпретатор, конвертор.
6. Первое поколение языков. Фортран, Алгол, Кобол
7. Интерактивное программирование. Бейсик.
8. Структурированное программирование и его история.
9. Теорема структурного программирования. Доказательство в программных кодах.
10. Язык Си. Состав языка. Стандарт языка.
11. Языки программирования Си++.

12. Компоненты алгоритмического языка. Примеры.
13. Основные стандартные функции в алгоритмических языках.
14. Модули. Описание модулей. Использование модулей
15. Этапы создания структурной программы.
16. Технология структурного программирования. Критерии качества программы.
17. Правила программирования. Документирование программы.
18. Указатели. Динамические структуры.
19. Обработка исключений и финальных завершений в программных средах.
20. Методологии программирования.

Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся.

ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО
ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

«СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ»

Направление подготовки: 15.03.05 Конструкторско-технологическое
обеспечение машиностроительных производств

Направленность (профиль): Технология машиностроения

Уровень высшего образования: бакалавриат

Форма обучения: очная, заочная

Королев
2023

Общие положения

Цель дисциплины:

– формирование представления о современном подходе к решению различных задач прикладной области с помощью ЭВМ;

– подготовка бакалавров к использованию современных компьютерных технологий и алгоритмов в будущей производственно-технической, проектно-конструкторской и исследовательской деятельности.

Задачи дисциплины:

– ознакомление бакалавров по данному направлению обучения с основным инструментальными и программными средствами компьютерного решения задач;

– обучение технологии структурного и модульного программирования;

– приобретение бакалаврами навыков написания, отладки и тестирования, документирования надежных, безопасных и легко модифицируемых программ.

1. Указания по проведению практических занятий

Практическое занятие 1-2

Вид практического занятия: *смешанная форма практического занятия.*

Тема и содержание практического занятия: Введение в курс «Современные языки программирования». Понятие системы программирования

Цель работы: На примере доступной системы программирования (например, С/С++/Питон) изучить основные компоненты системы и интерфейса пользователя для решения задач, сформулированных на языке программирования, поддерживаемого системой.

Основные положения темы занятия:

1. Редактор
2. Транслятор. Виды трансляторов.
3. Компоновщик.
4. Библиотечные модули.

Вопросы для обсуждения:

1. Реализация языка программирования.
2. Современные системы программирования.
3. Основные компоненты среды программирования.
4. Современные парадигмы программирования.

Образовательные технологии: проектная.

Продолжительность занятия – **4 ч.**

Практическое занятие 3

Вид практического занятия: *подготовка доклада.*

Тема и содержание практического занятия: Классификация ЯП.

Цель работы: Самостоятельная работа студентов по заранее выданным вопросам и обсуждение представленного материала на занятии в форме семинара.

Основные положения темы занятия:

1. Язык программирования и его детализация предписаний.
2. Генеалогия современных языков программирования высокого уровня

Вопросы для обсуждения:

1. Классы языков
 2. Императивная парадигма
 3. Объектно-ориентированная парадигма
 4. Структурные языки программирования
- Образовательные технологии: проектная.

Продолжительность занятия – 2 ч.

Практическое занятие 4

Вид практического занятия: *смешанная форма практического занятия.*

Тема и содержание практического занятия: Основные принципы программирования на языках высокого уровня.

Цель работы: Получение практических навыков в постановке, моделировании и решении практических задач с помощью компьютера

Основные положения темы занятия:

1. Этапы решения задач на ЭВМ
2. Постановка задачи.
3. Выбор входных/выходных данных
4. Алгоритмизация

Вопросы для обсуждения:

1. Составление ТЗ – заключительный этап постановки задачи
2. Основные шаги при составлении модели решаемой задачи
3. Выбор способа алгоритмизации
4. Язык С/С++/Питон как инструмент структурированного программирования
5. Основные понятия в алгоритмических языках.
6. Основные стандартные функции в алгоритмических языках.
7. Примеры записей арифметических и логических выражений на языке С/С++/Питон.

Образовательные технологии: проектная.

Продолжительность занятия – 2 ч.

Практическое занятие 5

Вид практического занятия: *смешанная форма практического занятия.*

Тема и содержание практического занятия: Технология структурного и модульного программирования.

Цель работы: Получение практических навыков в составлении алгоритмов решения различной сложности прикладных задач с последующей отладкой и тестированием написанного кода в соответствующей среде программирования.

Основные положения темы занятия:

1. Основные типы данных
2. Базовые алгоритмические структуры.
3. Процедуры и функции

Вопросы для обсуждения:

1. Простые и структурированные типы данных C/C++/Питон
2. Простые и структурные операторы C/C++/Питон
3. Процедуры передачи управления.
4. Совместимость типов.
5. Подпрограммы. Процедуры и функции. Примеры записи на C/C++/Питон
6. Глобальные и локальные переменные.
7. Структура исполняемой программы в оперативной памяти.
8. Виды параметров подпрограмм.

Образовательные технологии: проектная.

Продолжительность занятия – 2 ч.

Практическое занятие 6

Вид практического занятия: *смешанная форма практического занятия.*

Тема и содержание практического занятия: Стандартные библиотеки и их подпрограммы

Цель работы: Научиться использовать стандартные модули среды программирования C++ и Питон.

Основные положения темы занятия:

1. Создание модулей
2. Структура модулей
3. Содержание и состав стандартных модулей среды программирования C/C++ (WebKit, Libcurl, libjpg, DynaPDF, Libpq) и Питон (Dash, Pillow, Colorama, JMESPath, Simplejson, Flask, Beautiful soup)

Вопросы для обсуждения:

1. Преимущества модульного программирования
2. Содержание и состав стандартных модулей
3. C/C++ (WebKit, Libcurl, libjpg, DynaPDF, Libpq)
4. Питон (Dash, Pillow, Colorama, JMESPath, Simplejson, Flask, Beautiful soup)

Образовательные технологии: проектная.

Продолжительность занятия – 2 ч.

Практическое занятие 7

Вид практического занятия: *подготовка доклада.*

Тема и содержание практического занятия: Технология ООП

Цель работы: Самостоятельная работа студентов по заранее выданным вопросам и обсуждение представленного материала на занятии в форме семинара.

Основные положения темы занятия:

1. Основные принципы объектно-ориентированного программирования.
2. Вычислительная модель.

Вопросы для обсуждения:

1. Объекты. Данные (поля) и методы.
2. Инкапсуляция. Использование кода.
3. Наследование. Библиотеки классов.
4. Полиморфизм.
5. Языки и среды ООП

Образовательные технологии: проектная.

Продолжительность занятия – **2 ч.**

Практическое занятие 8

Вид практического занятия: *подготовка доклада.*

Тема и содержание практического занятия: Языки и системы функционального и логического программирования

Цель работы: Самостоятельная работа студентов по заранее выданным вопросам и обсуждение представленного материала на занятии в форме семинара.

Основные положения темы занятия:

1. Вычислительная модель функционального программирования.
2. Вычислительная модель логического программирования

Вопросы для обсуждения:

1. Языки и среды функционального программирования
2. Языки и среды логического программирования

Образовательные технологии: проектная.

Продолжительность занятия – **2 ч.**

3. Указания по проведению лабораторного практикума

Не предусмотрено учебным планом.

4. Указания по проведению самостоятельной работы студентов

Цель самостоятельной работы: подготовить бакалавров к самостоятельному научному творчеству.

Задачи самостоятельной работы:

- расширить знания по современным языкам и средам программирования;
- овладеть технологией модульного и структурного программирования;
- систематизировать знания, подходы, методы по решению прикладных задач пользователя с помощью ПК.

Объем времени на самостоятельную работу, и виды самостоятельной работы представлены в таблице 1.

Объем времени и виды самостоятельной работы

Виды самостоятельной работы	Очная форма обучения
	Всего академических часов – 108
Всего часов на самостоятельную работу	44 / 92
Вопросы, выносимые на самостоятельное изучение	10 / 22
Подготовка к практическим занятиям	10 / 22
Подготовка докладов	10 / 22
Выполнение практических заданий	14 / 26

Вопросы, выносимые на самостоятельное изучение:

1. Исследования современных тенденций в возникновении, развитии и использовании языков программирования для решения современных технико-экономических прикладных задач в различных вычислительных системах.

2. Исследование информации различной структуры и обработка её средствами среды программирования C++ и Питон.

3. Решение практических задач методом процедурной декомпозиции. Использование функций и процедур библиотек, C++ и Питон.

Тематическое содержание самостоятельной работы представлено в таблице 2.

Тематическое содержание самостоятельной работы

№ п.п.	Виды самостоятельной работы	Количество часов	Перечень заданий
1.	Вопросы, выносимые на самостоятельное изучение	10 / 22	Изучение открытых источников
2.	Подготовка к практическим занятиям	10 / 22	Изучение открытых источников при подготовке доклада на выбранную тему.
3.	Тематика докладов	10 / 22	1. Классы языков 2. Генеалогия современных языков программирования высокого уровня.
4.	Выполнение практических заданий	14 / 26	Построение математических моделей исследуемых объектов и систем; решение формализованных задач в соответствующих средах на ПК.

Примерные темы докладов

1. Язык программирования Perl
2. Язык программирования PHP и его реализации
3. Язык программирования РЕФАЛ
4. Язык программирования Python
5. Язык программирования Prolog
6. Язык программирования Ruby
7. Язык программирования Scala
8. Язык программирования Оберон
9. Язык программирования Симула

5. Указания по проведению контрольных работ для обучающихся очной и заочной формы обучения

5.1. Требования к структуре

Структура контрольной работы должна способствовать раскрытию темы: иметь титульный лист, содержание, введение, основную часть, заключение, список литературы.

5.2. Требования к содержанию

5.2.1. Во введении обосновывается актуальность темы, определяется цель работы, задачи и методы исследования.

5.2.2. При определении целей и задач исследования необходимо правильно их формулировать. Так, в качестве цели не следует употреблять глагол «сделать». Правильно будет использовать глаголы: «раскрыть», «определить», «установить», «показать», «выявить» и т.д.

5.2.3. Основная часть работы включает вопросы, каждый из которых посвящается решению задач, сформулированных во введении, и заканчивается констатацией итогов.

5.2.4. Приветствуется иллюстрация содержания работы таблицами, графическим материалом (рисунками, схемами и т.п.).

5.2.5. Необходимо давать ссылки на используемую Вами литературу.

5.2.6. Заключение должно содержать сделанные автором работы выводы, итоги исследования.

5.2.7. Вслед за заключением идет список литературы, который должен быть составлен в соответствии с установленными требованиями. Если в работе имеются приложения, они оформляются на отдельных листах, и должны быть соответственно пронумерованы.

5.3. Требования к оформлению.

Объём контрольной работы – 5...10 страниц формата А4, напечатанного с одной стороны текста (1,5 интервал, шрифт Times New Roman). Контрольная работа должна быть также представлена в электронном виде.

6. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Бедердинова О.И. Программирование на языках высокого уровня: учеб. пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. – М.: ИНФРА-М, 2019. – 159 с. – Текст: электронный.
- URL: <https://znanium.com/catalog/product/1044396>
- Режим доступа: по подписке.
2. Основы алгоритмизации и программирования на Python: учеб. пособие / С.Р. Гуриков. – М.: ФОРУМ: ИНФРА-М, 2017. – 343 с. — (Высшее образование: Бакалавриат).
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=924699>
- Режим доступа: по подписке.
3. Программирование на языке высокого уровня. Программирование на языке C++: учеб. пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; под ред. Л.Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2018. – 512 с.
- URL: <http://znanium.com/catalog.php?bookinfo=918098>
- Режим доступа: по подписке.
4. Шелудько В.М. Язык программирования высокого уровня Python: функции, структуры данных, дополнительные модули / В.М. Шелудько; Министерство науки и высшего образования РФ, Южный федеральный университет, Институт компьютерных технологий и информационной безопасности. – Ростов-на-Дону; Таганрог: Южный федеральный университет, 2017. – 108 с.
- URL: <http://biblioclub.ru/index.php?page=book&id=500060>
- Режим доступа: по подписке.

Дополнительная литература:

1. Колдаев В.Д. Численные методы и программирование: учеб. пособие / В.Д. Колдаев; под ред. проф. Л.Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2017. – 336 с.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=672965>
- Режим доступа: по подписке.
2. Программирование графики на C++. Теория и примеры: учеб. пособие / В.И. Корнеев, Л.Г. Гагарина, М.В. Корнеева. — М.: ИД «ФОРУМ»: ИНФРА-М, 2017. — 517 с. + Доп. материалы [Электронный ресурс; Режим доступа <http://www.znanium.com>]. – (Высшее образование). – www.dx.doi.org/10.12737/23113.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=562914>
- Режим доступа: по подписке.
3. Основы алгоритмизации и программирования: учебное пособие / под ред. проф. Л. Г. Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2017. – 416 с.
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=902236>
- Режим доступа: по подписке.

4. Программирование на С++ с погружением: практические задания и примеры кода - М.: НИЦ ИНФРА-М, 2016. - 80 с.: 60x90 1/16
- URL: <http://znanium.com/catalog.php?item=bookinfo&book=563294>
- Режим доступа: по подписке.

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

Интернет-ресурсы:

1. <http://biblioclub.ru/> - Университетская библиотека ONLINE
2. <http://www.znanium.com/> - Электронно-библиотечная система
3. <http://www.en.edu.ru/> - естественнонаучный образовательный портал
4. <http://www.academy.it.ru/> - академия АЙТИ.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения:

NotePad++, MSOffice, GCC + MinGW, Python 3, PyCharm.

Информационные справочные системы:

3. Ресурсы информационно-образовательной среды Университета.
4. Рабочая программа и методическое обеспечение по дисциплине «Современные языки программирования».