

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Воронежский государственный технический университет»

УТВЕРЖДАЮ  
Декан ФМАТ  
В.И. Рязских  
« 21 » 02 2023 г.  
Воронежский государственный технический университет  
Факультет  
машиностроения и  
аэрокосмической  
техники

**РАБОЧАЯ ПРОГРАММА**  
дисциплины (модуля)  
«Информатика»

**Направление подготовки** 15.03.05 – Конструкторско-технологическое  
обеспечение машиностроительных производств  
**Профиль** Металлообрабатывающие станки и комплексы  
**Квалификация выпускника** Бакалавр  
**Нормативный период обучения** 4 года / 4 г. и 11 м.  
**Форма обучения** Очная / Заочная  
**Год начала подготовки** 2023 г.

Автор программы \_\_\_\_\_ / Д. Е. Пачевский /

Заведующий кафедрой  
компьютерных интеллектуальных  
технологий проектирования \_\_\_\_\_ / М. И. Чижов /

Руководитель ОПОП \_\_\_\_\_ / М.Н. Краснова /

**Воронеж 2023**

# 1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

## 1.1. Цели дисциплины

- освоение методов решения инженерных задач с применением современных языков программирования;
- изучение технологий создания программного обеспечения на языке программирования Python при решении прикладных задач в области машиностроительного производства.

## 1.2. Задачи освоения дисциплины

- овладеть основами алгоритмизации и разработки программного обеспечения применительно к машиностроительному производству на языке программирования Python;
- применять интегрированные среды разработки для создания программного обеспечения с учетом особенностей технологий функционального и объектно-ориентированного программирования.

# 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина «Конструкторско-технологическая информатика» относится к дисциплинам обязательной части блока Б.1 учебного плана.

# 3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс изучения дисциплины «Конструкторско-технологическая информатика» направлен на формирование следующих компетенций:

ОПК-6 – Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.

Компетенция	Результаты обучения, характеризующие сформированность компетенции
ОПК-6	<b>знать</b> стандартные программные средства для решения задач в области конструкторско-технологического обеспечения машиностроительных производств; типы и структуры данных, используемые в языке программирования Python
	<b>уметь</b> решать задачи в области конструкторско-технологического обеспечения машиностроительных производств, используя язык программирования Python.
	<b>владеть</b> навыками разработки и использования современных информационных технологий, прикладных программных средств при решении задач в области конструкторско-технологического обеспечения машиностроительных производств.

#### 4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины «Конструкторско-технологическая информатика» составляет 5 зачетных единиц.

Распределение трудоемкости дисциплины по видам занятий

##### Очная форма обучения

Вид учебной работы	Всего часов	Семестры			
		2			
<b>Аудиторные занятия (всего)</b>	72	72			
В том числе:					
Лекции	36	36			
Практические занятия (ПЗ)	-	-			
Лабораторные работы (ЛР)	36	36			
<b>Самостоятельная работа</b>	63	63			
Курсовая работа (есть, нет)	есть	есть			
Контрольная работа (есть, нет)	нет	нет			
Вид промежуточной аттестации: экзамен	45	45			
Общая трудоемкость	час	180	180		
	зач. ед.	5	5		

##### Заочная форма обучения

Вид учебной работы	Всего часов	Семестры			
		4			
<b>Аудиторные занятия (всего)</b>	16	16			
В том числе:					
Лекции	4	4			
Практические занятия (ПЗ)	-	-			
Лабораторные работы (ЛР)	12	12			
<b>Самостоятельная работа</b>	155	155			
Курсовая работа (есть, нет)	есть	есть			
Контрольная работа (есть, нет)	нет	нет			
Вид промежуточной аттестации: экзамен	9	9			
Общая трудоемкость	час	180	180		
	зач. ед.	5	5		

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 5.1 Содержание разделов дисциплины и распределение трудоемкости по видам занятий

#### Очная форма

№ п/п	Наименование тем	Содержание разделов и тем	Лекции	Прак зан.	Лаб. зан.	СРС	Всего, час
<b>Разд.1 Основы синтаксиса Python</b>							
1	Введение в язык программирования.	Интегрированная среда программирования. Основные функции и особенности создания программы Самостоятельная работа: <i>Системы счисления. Позиционные и непозиционные системы. Использование двоичной, восьмеричной и шестнадцатеричной систем счисления.</i>	2	-	4	3	9
2	Переменные в Python	Концепция присваивания в Python. Динамическая типизация. Именованные переменные. Арифметические операции. Операции сравнения. Условная конструкция if. Самостоятельная работа: <i>Виды переменных и типы данных в Python при решении инженерных задач</i>	2	-	4	3	9
3	Циклы.	Цикл while. Цикл for. Функция range(). Самостоятельная работа: <i>Перевод чисел из одной системы в другую. Арифметические действия в различных системах счисления.</i>	2	-	4	3	9
4	Логические операции в Python	Основы алгебры логики. Таблицы истинности. Логические операции. Каскадные условные конструкции. Самостоятельная работа: <i>Системы счисления. Позиционные и непозиционные системы. Использование двоичной, восьмеричной и шестнадцатеричной систем счисления.</i>	2	-	4	3	9
5	Типы данных.	Типы данных при решении конструкторско-технологических задач. Тип данных bool. Константы True, False. Логические операции в Python.	2	-	-	3	5

		Самостоятельная работа: <i>Законы логических операций.</i>					
6	Специальные типы данных	Кортежи. Множества. Преобразование типов данных. Самостоятельная работа: <i>Явное и неявное приведение типов</i>	2	-	-	3	5
7	Списки.	Работа со списками. Перебор с помощью цикла for. Методы при работе со списками. Функции при работе со списками. Срезы. Самостоятельная работа: <i>Многомерные списки, Генераторы списков.</i>	2	-	-	4	6
<b>Разд.2 Основы программирования на Python.</b>							
8	Функции	Определение функции. Аргументы функции. Аргументы по умолчанию. Самостоятельная работа: <i>Генераторы. Декораторы функций.</i>	2	-	2	3	7
9	Функции с переменным числом аргументов	Переменное число аргументов. Именованные аргументы. Условия именованности. Дополнительные именованные аргументы. Возвращаемые значения. Правила видимости. Инструкция global. Анонимные функции. Самостоятельная работа: <i>Генераторы. Декораторы функций.</i>	2	-	4	3	9
10	Работа со строками.	Строки. Основные функции и методы строк. Форматирование строк. Оператор %. Спецификаторы и модификаторы. Самостоятельная работа: <i>Словари. Работа со словарями.</i>	2	-	2	3	7
11	Объектно-ориентированное программирование	Базовые понятия ООП: объект, атрибуты и методы, класс. Самостоятельная работа: <i>Рекурсия. Факториал числа.</i>	2	-	-	4	6
12	Принципы объектно-ориентированного программирования	Инкапсуляция. Наследование. Полиморфизм. Самостоятельная работа: <i>Рекурсия. Факториал числа.</i>	2	-	-	4	6
13	Модули.	Импорт модулей. Стандартная библиотека Python. Модуль random. Исключения. Обработка исключений. Самостоятельная работа: <i>Файлы. Работа с файлами.</i>	2	-	2	4	8

14	Прикладные библиотеки Python.	Введение в массивы библиотеки NumPy. Срезы массивов. Изменения формы массивов. Слияние и разбиение массивов. Самостоятельная работа: <i>Пользовательские настройки графиков. Множественные субграфики.</i>	2	-	2	4	8
15	Визуализация данных при решении конструкторско-технологических задач	Визуализация с помощью библиотеки Matplotlib. Простые линейные графики. Самостоятельная работа: <i>Пользовательские настройки графиков. Множественные субграфики.</i>	2	-	2	4	8
16	Технологии разработки инженерных задач	Жизненный цикл программного обеспечения. Технологии и примеры разработки программного обеспечения решения конструкторско-технологических задач	2	-	2	4	8
17	Разработка графического интерфейса пользователя	Средства общения с ЭВМ. Обработка событий с мыши и клавиатуры. Разработка графического окна для программы	2	-	2	4	8
18	Технологии создания web-приложений	Виды и возможности web-приложений. Организация обмена данными «клиент-сервер». Технологии разработки web-приложений	2	-	2	4	8
		<i>Итого</i>	36	-	36	63	135
		<i>Экзамен</i>	-	-	-	-	45
		<b>Всего</b>	<b>36</b>	<b>-</b>	<b>36</b>	<b>63</b>	<b>180</b>

### Заочная форма обучения

№ п/п	Наименование тем	Содержание разделов и тем	Лекции	Прак зан.	Лаб. зан.	СРС	Всего, час
<b>Разд.1 Основы синтаксиса Python</b>							
1	Введение в язык программирования.	Интегрированная среда программирования. Основные функции и особенности создания программы Самостоятельная работа: <i>Системы счисления. Позиционные и непозиционные системы. Использование двоичной, восьмеричной и шестнадцатеричной систем счисления.</i>	-	-	-	9	9,0

2	Переменные в Python	Концепция присваивания в Python. Динамическая типизация. Именованные переменные. Арифметические операции. Операции сравнения. Условная конструкция if. Самостоятельная работа: <i>Виды переменных и типы данных в Python при решении инженерных задач</i>	0,5	-	1	9	10,5
3	Циклы.	Цикл while. Цикл for. Функция range(). Самостоятельная работа: <i>Перевод чисел из одной системы в другую. Арифметические действия в различных системах счисления.</i>		-	1	9	10
4	Логические операции в Python	Основы алгебры логики. Таблицы истинности. Логические операции. Каскадные условные конструкции. Самостоятельная работа: <i>Системы счисления. Позиционные и непозиционные системы. Использование двоичной, восьмеричной и шестнадцатеричной систем счисления.</i>	-	-	1	9	10
5	Типы данных.	Типы данных при решении конструкторско-технологических задач. Тип данных bool. Константы True, False. Логические операции в Python. Самостоятельная работа: <i>Законы логических операций.</i>	0,5	-	1	9	10,5
6	Специальные типы данных	Кортежи. Множества. Преобразование типов данных. Самостоятельная работа: <i>Явное и неявное приведение типов</i>		-	-	8	8
7	Списки.	Работа со списками. Перебор с помощью цикла for. Методы при работе со списками. Функции при работе со списками. Срезы. Самостоятельная работа: <i>Многомерные списки, Генераторы списков.</i>	-	-	1	8	9
<b>Разд.2 Основы программирования на Python.</b>						-	
8	Функции	Определение функции. Аргументы функции. Аргументы по умолчанию. Самостоятельная работа: <i>Гене-</i>	-	-	1	8	9

		<i>раторы. Декораторы функций.</i>					
9	Функции с переменным числом аргументов	Переменное число аргументов. Именованные аргументы. Условия именованности. Дополнительные именованные аргументы. Возвращаемые значения. Правила видимости. Инструкция global. Анонимные функции. Самостоятельная работа: <i>Генераторы. Декораторы функций.</i>	0,5	-	1	8	9,5
10	Работа со строками.	Строки. Основные функции и методы строк. Форматирование строк. Оператор %. Спецификаторы и модификаторы. Самостоятельная работа: <i>Словари. Работа со словарями.</i>	0,5	-	-	8	8,5
11	Объектно-ориентированное программирование	Базовые понятия ООП: объект, атрибуты и методы, класс. Самостоятельная работа: <i>Рекурсия. Факториал числа.</i>	0,5	-	1	8	9,5
12	Принципы объектно-ориентированного программирования	Инкапсуляция. Наследование. Полиморфизм. Самостоятельная работа: <i>Рекурсия. Факториал числа.</i>	0,5	-	-	8	8,5
13	Модули.	Импорт модулей. Стандартная библиотека Python. Модуль random. Исключения. Обработка исключений. Самостоятельная работа: <i>Файлы. Работа с файлами.</i>	0,5	-	1	9	10,5
14	Прикладные библиотеки Python.	Введение в массивы библиотеки NumPy. Срезы массивов. Изменения формы массивов. Слияние и разбиение массивов. Самостоятельная работа: <i>Пользовательские настройки графиков. Множественные субграфики.</i>	0,5	-	1	9	10,5
15	Визуализация данных при решении конструкторско-технологических задач	Визуализация с помощью библиотеки Matplotlib. Простые линейные графики. Самостоятельная работа: <i>Пользовательские настройки графиков. Множественные субграфики.</i>	-	-	1	9	10



16	Технологии разработки инженерных задач	Жизненный цикл программного обеспечения. Технологии и примеры разработки программного обеспечения решения конструкторско-технологических задач	-	-	1	9	10
17	Разработка графического интерфейса пользователя	Средства общения с ЭВМ. Обработка событий с мыши и клавиатуры. Разработка графического окна для программы	-	-	-	9	9
18	Технологии создания web-приложений	Виды и возможности web-приложений. Организация обмена данными «клиент-сервер». Технологии разработки web-приложений	-	-	-	9	9
		<i>Итого</i>	4	-	12	155	171
		<i>Экзамен</i>	-	-	-	-	9
		<b>Всего</b>	<b>4</b>	<b>-</b>	<b>12</b>	<b>155</b>	<b>180</b>

## 5.2 Перечень лабораторных работ

1. Интегрированная среда разработки.
2. Алгоритмы на Python.
3. Работа с файлами и файловой системой.
4. Обработка и визуализация данных на языке Python.

## 6 ПРИМЕРНАЯ ТЕМАТИКА КУРСОВЫХ ПРОЕКТОВ (РАБОТ) И КОНТРОЛЬНЫХ РАБОТ

В соответствии с учебным планом освоение дисциплины предусматривает выполнение курсовой работы во 2 семестре для очной формы обучения и в 4 семестре для заочной формы обучения.

Основная тематика курсовой работы – «Автоматизация технологической подготовки производства».

Задачи, решаемые при выполнении курсовой работы:

- решение конкретных конструкторско-технологических задач технологической подготовки производства к выпуску машиностроительного изделия с использованием языка программирования Python;
- выполнение расчета режимных параметров технологических операций с использованием языка программирования Python;
- разработка программы для расчета режимных параметров технологической операции на языке программирования Python;
- получение навыков самостоятельной работы со стандартами, справочной литературой, каталогами, справочниками.

Курсовая работа состоит из расчетно-пояснительной записки, которая должна содержать следующие разделы:

Введение

1 Теоретический вопрос

2 Исходные данные. Входные и выходные параметры расчета

3 Алгоритм расчета

4 Текст программы

5 Результаты работы

Заключение

Список литературы

Приложения

Для групп профиля «Металлообрабатывающие станки и комплексы» рекомендуются следующие темы курсовой работы:

1. Автоматизация расчета скорости резания при продольном точении
2. Автоматизация расчета скорости резания при поперечном точении
3. Автоматизация расчета скорости резания при отрезании
4. Автоматизация расчета скорости резания при нарезании резьбы
5. Автоматизация расчета скорости резания при фасонном точении
6. Автоматизация расчета силы резания при продольном точении
7. Автоматизация расчета силы скорости и мощности резания при поперечном точении
8. Автоматизация расчета силы скорости и мощности резания при отрезании
9. Автоматизация расчета силы и мощности резания при нарезании
10. Автоматизация расчета силы и мощности резания при фасонном точении
11. Автоматизация расчета скорости резания при сверлении
12. Автоматизация расчета скорости резания при рассверливании
13. Автоматизация расчета скорости резания при зенкеровании
14. Автоматизация расчета скорости резания при развертывании
15. Автоматизация расчета крутящего момента и мощности резания при развертывании
16. Автоматизация расчета подачи и скорости резания при фрезеровании
17. Автоматизация расчета крутящего момента и мощности резания при фрезеровании
18. Автоматизация расчета скорости резания при нарезании крепежной резьбы резцами с пластинами из твердого сплава
19. Автоматизация расчета скорости резания при нарезании крепежной резьбы резцами из быстрорежущей стали
20. Автоматизация расчета скорости резания при вихревом нарезании метрической и трапецеидальной резьб твердосплавными резцами во вращающихся головках
21. Автоматизация расчета скорости резания при нарезании резьбы с ограниченным выходом резца
22. Автоматизация расчета тангенциальной составляющей силы резания при нарезании резьбы резцами
23. Автоматизация расчета крутящего момента при нарезании резьбы метчиками

24. Автоматизация расчета крутящего момента при нарезании резьбы резьбовыми головками

25. Автоматизация расчета мощности при нарезании резьбы резцами ([2], стр. 293)

26. Автоматизация расчета мощности при нарезании резьбы метчиками, плашками и резьбовыми головками

27. Автоматизация расчета мощности при вихревом нарезании треугольной резьбы

28. Автоматизация расчета мощности при вихревом нарезании трапецидальной резьбы

Учебным планом по дисциплине не предусмотрено выполнение контрольной работы (контрольных работ) во 2 семестре для очной формы обучения и в 4 семестре для заочной формы обучения.

## **7 ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

### **7.1 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания**

#### **7.1.1 Этап текущего контроля**

Результаты текущего контроля знаний и межсессионной аттестации оцениваются по следующей системе:

«аттестован»;

«не аттестован».

<b>Компетенция</b>	<b>Результаты обучения, характеризующие сформированность компетенции</b>	<b>Критерии оценивания</b>	<b>Аттестован</b>	<b>Не аттестован</b>
ОПК-6	<b>знать</b> стандартные программные средства для решения задач в области конструкторско-технологического обеспечения машиностроительных производств; типы и структуры данных, используемые в языке программирования Python	Активная работа на лабораторных занятиях, отвечает на вопросы при защите курсовой работы	Выполнение работ в сроки, предусмотренные в рабочей программе	Невыполнение работ в сроки, предусмотренное в рабочей программе.
	<b>уметь</b> решать задачи в области конструкторско-технологического обеспечения машиностроительных производств, используя язык программирования Python.	Активная работа на лабораторных занятиях, разработка и выполнение курсовой работы.	Выполнение работ в сроки, предусмотренные в рабочей программе	Невыполнение работ в сроки, предусмотренное в рабочей программе
	<b>владеть</b> навыками разработки и использования современных информационных технологий, прикладных программных	Защита лабораторных работ и защита курсовой работы	Выполнение работ в сроки, предусмотренные	Невыполнение работ в сроки, предусмотренное в рабочей

	средств при решении задач в области конструкторско-технологического обеспечения машиностроительных производств.		в рабочей программе	программе
--	---	--	---------------------	-----------

### 7.1.2 Этап промежуточного контроля знаний

Результаты промежуточного контроля знаний для очной формы обучения оцениваются в 2 семестре, для заочной формы обучения в 4 семестре по системе:

- «отлично»;
- «хорошо»;
- «удовлетворительно»;
- «неудовлетворительно»

Компетенция	Результаты обучения, характеризующие сформированность компетенции	Критерии оценивания	Отлично	Хорошо	Удовл	Неудовл
ОПК-6	знать стандартные программные средства для решения задач в области конструкторско-технологического обеспечения машиностроительных производств; типы и структуры данных, используемые в языке программирования Python	Тест	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	уметь решать задачи в области конструкторско-технологического обеспечения машиностроительных производств, используя язык программирования Python.	Тест, решение стандартных задач	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов
	владеть навыками разработки и использования современных информационных технологий, прикладных программных средств при решении задач в области конструкторско-	Тест, решение прикладных задач в предметной области	Выполнение теста на 90-100%	Выполнение теста на 80-90%	Выполнение теста на 70-80%	В тесте менее 70% правильных ответов

	технологического обеспечения машино-строительных производств.					
--	---	--	--	--	--	--

## 7.2 Примерный перечень оценочных средств (типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности)

### 7.2.1 Примерный перечень заданий для подготовки к тестированию

1. Что будет напечатано при исполнении следующего кода? Используется Python 3.x.

```
print(type(5/2))
```

1. type 'int'
2. type 'number'
3. type 'double'
4. type 'tuple'

2. Что выведет следующий фрагмент кода?

```
x = 4.5
y = 2
print(x // y)
```

1. 2.0
2. 2.25
3. 9.0
4. False

3. Что будет напечатано?

```
x = True
y = False
if not x:
    print(1)
elif not x and y:
    print(2)
elif not x or not y:
    print(3)
else:
    print(4)
```

1. 1
2. 2
3. 3
4. 4

4. Что будет напечатано?

```
x = "summer sun"
```

```
print("%s" % x[3:5])
```

1. mer
2. mme
3. me
4. Syntax Error

5. Что выведет следующий код?

```
d = lambda p: p * 2
```

```
t = lambda p: p * 3
```

```
x = 2
```

```
x = d(x)
```

```
x = t(x)
```

```
x = d(x)
```

```
print(x)
```

1. 12
2. 24
3. 36
4. 48

6. Что делает следующий код?

```
def a(b, c, d): pass
```

1. Определяет список и инициализирует его.
2. Определяет функцию, которая ничего не делает.
3. Определяет функцию, которая передает параметры.
4. Определяет пустой класс.

7. Что будет напечатано?

```
dict = {"one", "two", "three", "four"}
```

```
print(dict['three'])
```

1. two
2. three
3. four
4. TypeError.

8. Что выведет следующая программа?

```
a = [1,2,3,None,(),[],]
```

```
print(len(a))
```

1. 4
2. 5
3. 6
4. 7

9. Что выведет следующий цикл?

```
for i in range(5):
```

```
    if i == 3:
```

```
        continue
```

*print(i)*

1. 0, 1, 2, 3, 4
2. 0, 1, 2, 4
3. 0, 1, 2, 4, 5
4. 1, 2, 3, 4, 5

10. Что выведет следующая программа?

```
a=[0, 1, 2]
```

```
b=a
```

```
b[0]=10
```

```
print(a)
```

1. [0, 1, 2]
2. [10, 1, 2]
3. [1, 2]
4. Type Error

### 7.2.2 Примерный перечень заданий к выполнению стандартных задач

1. В Python используется:

- 1) Статическая типизация
- 2) Динамическая типизация
- 3) Базовая типизация
- 4) Интегрированная типизация

2. Объектно-ориентированное программирование не использует следующую концепцию:

- 1) Инкапсуляция
- 2) Наследование
- 3) Полиморфизм
- 4) Релятивизм

3. Инкапсуляция присуща:

- 1) Объектно-ориентированному программированию
- 2) Статическому программированию
- 3) Линейному программированию
- 4) Структурному программированию

4. Для импорта модуля в Python используется:

- 1) import
- 2) add
- 3) additional
- 4) export

5. Для форматирования строк в Python используется:

- 1) Оператор %
- 2) Оператор \*

- 3) Оператор \$
- 4) Оператор #

6. Для создания списка в Python используется функция

- 1) set()
- 2) list()
- 3) dict()
- 4) tuple()

7. Для работы с менеджером контекста в Python используется ключевое слово:

- 1) with
- 2) work
- 3) where
- 4) why

8. Декоратор функции в Python позволяет:

- 1) Модифицировать ее поведение
- 2) Удалить функцию
- 3) Типизировать функцию
- 4) Использовать функцию в качестве класса

9. lambda функции в Python – это:

- 1) статические функции
- 2) динамические функции
- 3) анонимные функции
- 4) линейные функции

10. bool в Python – это тип данных:

- 1) вещественный
- 2) строковый
- 3) списковый
- 4) логический

### 7.2.3 Примерный перечень заданий к выполнению прикладных задач

1. Найти уравнение прямой, проходящей через две точки.

На вход подаются координаты  $x_1, y_1, x_2, y_2$  через пробел. На выходе должна выводиться функция прямой, проходящей через заданные точки.

*in: 0 1 3 4*

*out:  $y = 1.00*x + 1.00$*

2. Найти площадь полной поверхности цилиндра.

На вход подается радиус цилиндра и его высота. На выходе имеем площадь полной поверхности цилиндра.

*in: 1*

*in: 10*



*out: 69.12*

3. Определить, в какой четверти находится точка.

На вход подаются координаты точки. На выходе выводится номер четверти, в которой располагается данная точка.

*in: -1*

*in: 10*

*out: Точка в II четверти*

4. Определить количество четных и нечетных цифр в числе.

На вход подается случайное число произвольной длины. На выход выводится количество четных и нечетных цифр, содержащихся в числе.

*in: 1122*

*out: Четных: 2*

*out: Нечетных: 2*

5. Найти максимальную цифру случайного числа.

На вход подается случайное число произвольной длины. На выход выводится самая большая цифра в числе.

*in: 1123.756*

*out: 7*

6. Напишите программу, которая считывает целые числа с консоли, по одному числу в строке.

Для каждого введенного числа проверить:

если число меньше 10, то пропускаем это число;

если число больше 100, то прекращаем считывать числа.

В остальных случаях вывести это число обратно на консоль в отдельной строке

*in: 55*

*9*

*46*

*111*

*out: 55*

*46*

7. Вычислить площадь треугольника, используя формулу Герона.

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

где  $p = (a+b+c)/2$  – полупериметр треугольника.

На вход программе подаются целые числа, выводом программы должно являться вещественное число, соответствующее площади треугольника

*in: 3*

*4*

*5*

*out: 6*

8. Напишите программу, принимающую на вход целое число, которая выводит True, если переданное значение попадает в интервал  $(-15,12] \cup (14,17) \cup [19,+\infty)$  и False в противном случае (регистр символов имеет значение).

*in: 15*

*out: True*

*in: 17*

*out: False*

9. Напишите программу, которая считывает со стандартного ввода целые числа, по одному числу в строке, и после первого введенного нуля, выводит сумму полученных на вход чисел.

*in: 3*

*4*

*5*

*0*

*out: 12*

10. Напишите программу, на вход которой подается одна строка с целыми числами.

Программа должна вывести сумму этих чисел.

*in: 1 2 3 4 5 6 7 8 9*

*out: 45*

#### **7.2.4 Примерный перечень вопросов для подготовки к зачету**

Не предусмотрено учебным планом

#### **7.2.5 Примерный перечень вопросов для подготовки к экзамену**

1. Предпосылки автоматизации проектирования машин
2. Инженерные расчеты в конструкторско-технологических задачах
3. Языки программирования, применяемые при решении конструкторско-технологических задач
4. Интегрированные среды разработки программного обеспечения и области их применения
5. Работа в IDE Microsoft Visual Studio
6. Легковесные редакторы программного кода: Visual Studio Code
7. Структура программ и применение переменных при решении конструкторско-технологических задач
8. Типы данных и их применение при решении конструкторско-технологических задач
9. Операторы, используемые при решении конструкторско-технологических задач
10. Назначение и виды условных операторов

11. Операторы управления и логические операторы
12. Операторы приведения типов данных в современных языках программирования
13. Процедуры и функции
14. Виды и операторы циклических процессов
15. Применение библиотек процедур при решении конструкторско-технологических задач
16. Массивы при решении инженерных задач
17. Переменные, операторы и методы создания массива данных
18. Двумерные массивы
19. Средства общения с ЭВМ
20. Обработка событий клавиатуры в инженерных приложениях
21. Обработка событий мыши в инженерных приложениях
22. Жизненный цикл программного обеспечения
23. Технологии разработки программного обеспечения
24. Примеры и возможности инженерных приложений

#### **7.2.6 Методика выставления оценки при проведении промежуточной аттестации**

Промежуточная аттестация проводится в соответствии с календарным графиком; учебным планом при промежуточной аттестации по дисциплине предусмотрен экзамен.

К промежуточной аттестации допускаются обучающиеся, получившие положительную оценку по выполненным лабораторным работам, при защите курсовой работы и по результатам текущей аттестации.

Промежуточная аттестация проводится путем организации опроса в устной и письменной форме с использованием прикладных программных средств по тестам, каждый из которых содержит 5 тестовых заданий, 5 стандартных задач и 5 прикладных задач. Каждый правильный ответ на вопрос тестового задания оценивается 2 баллами, каждая правильно решенная стандартная задача оценивается 2 баллами и каждая прикладная задача оценивается 2 баллами. Наибольшее количество набранных баллов – 30.

По результатам экзамена студентам ставятся оценки:

1. Оценка «Неудовлетворительно» ставится в случае, если студент набрал менее 16 баллов.
2. Оценка «Удовлетворительно» ставится в случае, если студент набрал от 16 до 20 баллов.
3. Оценка «Хорошо» ставится в случае, если студент набрал от 21 до 25 баллов.
4. Оценка «Отлично» ставится, если студент набрал от 26 до 30 баллов.

Во время защиты курсовой работы, обучающийся должен обосновать предложения по ее разработке и, для автоматизации заданного технологического процесса, разработать программное обеспечение. По результатам защиты курсовой работы обучающимся ставятся оценки:

1. «Отлично» – при правильном выполнении всех пунктов курсовой работы;
2. «Хорошо» – при выполнении только расчетной части и графического плана;
3. «Удовлетворительно» – при выполнении только расчетной части;
4. «Неудовлетворительно» – при невыполнении графической и расчетной части.

### 7.2.7 Паспорт оценочных материалов

№ п/п	Контролируемые темы дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	Введение в язык программирования.	ОПК-6	Тест, экзамен, устный опрос
2	Переменные в Python	ОПК-6	Тест, экзамен, устный опрос
3	Циклы.	ОПК-6	Тест, экзамен, устный опрос
4	Логические операции в Python	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
5	Типы данных.	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
6	Специальные типы данных	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
7	Списки.	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
8	Функции	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
9	Функции с переменным числом аргументов	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
10	Работа со строками.	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
11	Объектно-ориентированное программирование	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
12	Принципы объектно-ориентированного программирования	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
13	Модули.	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
14	Прикладные библиотеки Python.	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
15	Визуализация данных при решении конструкторско-технологических задач	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
16	Технологии разработки инженерных задач	ОПК-6	Тест, экзамен, устный опрос, курсовая работа, защита
17	Разработка графического интерфейса пользователя	ОПК-6	Тест, экзамен, устный опрос
18	Технологии создания web-приложений	ОПК-6	Тест, экзамен, устный опрос

### **7.3 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Проверка знаний на лабораторных занятиях, которая проводится в форме фронтального устного опроса, фиксируется преподавателем и доводится до сведения каждого обучающегося. Правильность выполнения лабораторной работы характеризует практическую освоенность материала по ее теме.

Тестирование осуществляется либо при помощи компьютерной системы тестирования, либо с использованием тестовых заданий на бумажном носителе. Время тестирования 30 мин. Затем экзаменатором осуществляется проверка теста, и выставляется оценка согласно методике выставления оценки при проведении промежуточной аттестации.

Решение стандартных задач осуществляется либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время подготовки их решения - 30 мин. Затем экзаменатором осуществляется проверка решения стандартных задач и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Решение прикладных задач осуществляется либо при помощи компьютерной системы тестирования, либо с использованием выданных задач на бумажном носителе. Время подготовки их решения - 30 мин. Затем экзаменатором осуществляется проверка решения прикладных задач и выставляется оценка, согласно методике выставления оценки при проведении промежуточной аттестации.

Защита курсовой работы осуществляется согласно требованиям, предъявляемым к работе, описанным в методических материалах. Примерное время защиты на одного обучающегося составляет 20 мин.

## **8 УЧЕБНО МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **8.1 Перечень учебной литературы, необходимой для освоения дисциплины**

1. Чижов, М.И. Информатика и информационные системы [Электронный ресурс]: конспект лекций по дисциплине «Информатика»: учеб. пособие / М. И. Чижов, А.Н. Юров. – Электрон. текстовые, граф. дан. – Воронеж: ВГТУ, 2003. 148 с. – 1 диск. – Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp/>

2. Уэс, Маккинли. Python и анализ данных [Текст]: учебник / Маккинли Уэс. – СПб.: Питер, 2010. – 576 с.

3. Алексеев, А.П. Информатика 2007 [Текст] / А.П. Алексеев. – М.: Солон-Пресс, 2007.– 608 с.

4. Острейковский, В.А. Информатика [Текст]: учеб. для вузов / В.А. Острейковский – М.: Высшая школа, 2007. – 511 с.

5. Чижов, М.И. МУ к выполнению лабораторных работ «Алгоритмический язык C<sup>++</sup> для создания консольных приложений» по дисциплине «Информатика» для студентов очной и очно-заочной форм обучения [Текст] / М.И. Чижов, А.Н. Юров. – Воронеж: ВГТУ, 2009. – Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp>.

**6. Методические рекомендации по выполнению курсовых проектов (работ)** по программам высшего образования – программам бакалавриата, специалитета, магистратуры / ФГБОУ ВО «Воронежский государственный технический университет»; сост.: В.Н. Почечихина, И.Н. Крючкова, Е.И. Головина. – Воронеж: изд-во ВГТУ, 2020. – 10 с.– Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp>. – Файл: [МР по проектированию.pdf](#).

7. Организация самостоятельной работы обучающихся: методические указания для студентов, осваивающих основные образовательные программы высшего образования – бакалавриата, специалитета, магистратуры: методические указания / сост. В.Н. Почечихина, И.Н. Крючкова, Е.И. Головина, В.Р. Демидов; ФГБОУ ВО «Воронежский государственный технический университет». – Воронеж, 2020. – 14 с. – Файл: OCP.PDF. – Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp>.

**8. Конструкторско-технологическая информатика:** методические указания к выполнению курсовой работы для студентов направления 15.03.05 «Конструкторско-технологическое обеспечение машиностроительных производств» (профили «Технология машиностроения», «Металлообрабатывающие станки и комплексы», «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства») всех форм обучения [Электронный ресурс] / сост.: С. Л. Новокщенов - Воронеж: ФГБОУ ВО «ВГТУ»; 2021. Изд. № 463-2021 – Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp>. – Файл: [МУ\\_КР\\_КТИ.pdf](#)

**9. Конструкторско-технологическая информатика:** методические указания к выполнению лабораторных работ для студентов, обучающихся по направлению 15.03.05 «Конструкторско-технологическое обеспечение машиностроительных производств» (профили «Технология машиностроения», «Металлообрабатывающие станки и комплексы» и «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства») всех форм обучения [Электронный ресурс] / сост.: С. Л. Новокщенов - Воронеж: ФГБОУ ВО «ВГТУ»; 2021. Изд. № 479-2021 – Режим доступа: <http://bibl.cchgeu.ru/MarcWeb2/Found.asp>– Файл: [МУ\\_ЛР\\_КТИ.pdf](#).

**8.2 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень лицензионного программного обеспечения, ресурсов информационно-телекоммуникационной сети «Интернет», современных профессиональных баз данных и информационных справочных систем**

## **Лицензионное программное обеспечение**

Adobe Acrobat Reader

Google Chrome

LibreOffice

WinDjView

Notepad++

Visual Studio Code

## **Ресурсы информационно-телекоммуникационной сети «Интернет»**

<http://www.edu.ru/>

Образовательный портал ВГТУ

## **Информационные справочные системы**

<http://window.edu.ru>

<https://wiki.cchgeu.ru/>

## **Современные профессиональные базы данных**

*Ресурс машиностроения*

Адрес ресурса: <http://www.i-mash.ru/>

*Машиностроение: сетевой электронный журнал*

Адрес ресурса: <http://indust-engineering.ru/archives-rus.html>

*Библиотека Машиностроителя*

Адрес ресурса: <https://lib-bkm.ru/>

## **9 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА**

Лекционные занятия и лабораторные работы проводятся в компьютерном классе корпуса № 1 кафедры АОМП 01.6/1, в котором находятся:

- компьютеры с программным оснащением для выполнения конструкторско-технологической документации по технологическим процессам КШО;
- интерактивная доска 78" ActivBoard 178, ПО ActivInspire; проектор; мультимедиа-проектор Sony VPL-SX125, ноутбук: фильмы, видеофильмы, видеофрагменты (графические файлы по всем лекционным темам для демонстрации слайдов непосредственно в лекционной аудитории);
- слайды, видеоматериалы по КШО и МРС.

## **10 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)**

По дисциплине «Конструкторско-технологическая информатика» читаются лекции, проводятся лабораторные занятия.

Основой изучения дисциплины являются лекции, на которых излагаются наиболее существенные и трудные вопросы, а также вопросы, не нашедшие отражения в учебной литературе.

Выполнение лабораторных работ направлено на приобретение практических навыков использования программного обеспечения на языке программирования Python для расчета режимных параметров технологических операций, других конструкторско-технологических задач подготовки производства к выпуску машиностроительного изделия. Занятия проводятся в аудитории, оснащенной компьютерами и необходимым программным обеспечением для выполнения заданий, поставленных условиями лабораторных работ.

Большое значение по закреплению и совершенствованию знаний имеет самостоятельная работа студентов, которая подкреплена учебниками, учебными пособиями, конспектами лекций, методическими разработками по курсовой работе, лабораторным работам, учебным программным обеспечением, справочной литературой и другими материалами.

Большое значение имеют консультации, проводимые преподавателем, как по курсовой работе, так и по теоретическим материалам дисциплины.

Контроль усвоения материала дисциплины проводится проверкой курсовой работы, защитой курсовой работы.

Освоение дисциплины оценивается на экзамене.

Вид учебных занятий	Деятельность студента
Лекция	<p>Написание конспекта лекций:</p> <ul style="list-style-type: none"> <li>- кратко, схематично, последовательно фиксировать основные положения, формулировки, обобщения, графики и схемы, выводы;</li> <li>- выделять важные мысли, ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначение вопросов, терминов, материала, которые вызывают трудности, поиск ответов в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на лабораторной работе.</li> </ul>
Лабораторные работы	<p>Перед каждой лабораторной работой студент должен ознакомиться с методическими указаниями, изучить теоретический материал и рекомендованную литературу к данной лабораторной работе, ознакомиться с ее организацией; уяснить цели задания, подготовиться и познакомиться с нормативной, справочной и учебной литературой и обратить внимание на рекомендации преподавателя: какие основные информационные данные извлечь из этих источников.</p>
Самостоятельная работа	<p>Самостоятельная работа студентов способствует глубокому усвоению учебного материала и развитию навыков самообразования. Самостоятельная работа предполагает следующие составляющие:</p> <ul style="list-style-type: none"> <li>- работа с текстами: учебниками, справочниками, дополнительной литературой, а также проработка конспектов лекций;</li> <li>- выполнение домашних заданий и расчетов;</li> <li>- работа над темами для самостоятельного изучения;</li> <li>- участие в работе студенческих научных конференций, олимпиад.</li> </ul>
Подготовка к	<p>При подготовке промежуточной аттестации по дисциплине необ-</p>



промежуточной аттестации по дисциплине	<p>ходимо ориентироваться на конспекты лекций, основную и рекомендуемую литературу, выполненные лабораторные работы.</p> <p>Работа студента при подготовке к аттестации должна включать: изучение учебных вопросов; распределение времени на подготовку; консультирование у преподавателя по трудно усвояемым вопросам; рассмотрение наиболее сложных из них в дополнительной литературе, или других информационных источниках, предложенных преподавателем.</p>
--	--

## ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п/п	Перечень вносимых изменений	Дата внесения изменений	Подпись заведующего кафедрой, ответственной за реализацию ОПОП
1			
2			
3			