

УТВЕРЖДАЮ  
Декан факультета

\_\_\_\_\_ Знаменский Е.А.

«\_\_\_» \_\_\_\_\_ 20\_\_

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

Направление/специальность подготовки	15.04.03 Прикладная механика
Специализация/профиль/программа подготовки	Механика процессов обработки давлением
Уровень высшего образования	Магистратура
Форма обучения	Очная
Факультет	Е Оружие и системы вооружения
Выпускающая кафедра	Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ СИСТЕМ
Кафедра-разработчик рабочей программы	Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ СИСТЕМ

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
5	10	3	108	34	0	0	34	74	0	0	74	зач.

*ЛИСТ СОГЛАСОВАНИЯ*

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО  
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

**15.04.03 Прикладная механика**

год набора группы: 2025

Программу составил:

Кафедра Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ  
СИСТЕМ \_\_\_\_\_

Сидоренко Тимофей Владимирович, старший преподаватель

Программа рассмотрена  
на заседании кафедры-разработчика  
рабочей программы **Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ СИСТЕМ**

Заведующий кафедрой Нестеров Н.И., к.т.н., доц. \_\_\_\_\_

Программа рассмотрена  
на заседании выпускающей кафедры

**Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ СИСТЕМ**

Заведующий кафедрой Нестеров Н.И., к.т.н., доц. \_\_\_\_\_

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ**

### **Разделы рабочей программы**

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### **Приложения к рабочей программе дисциплины**

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

## 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-12 — Способен создавать алгоритмы цифровой обработки баз данных результатов испытаний и эксплуатации сложных деталей и узлов в машиностроении, разрабатывать современные цифровые программы расчетов и проектирования деталей, узлов, конструкций, машин и материалов с учетом требований надежности, долговечности и безопасности их эксплуатации

Формированию компетенций служит достижение следующих результатов образования:

### **ОПК-12**

#### *знания:*

основные алгоритмические конструкции;

базовые алгоритмы обработки данных;

базовый синтаксис и основные структуры данных языка программирования Python;

представление о технологии программирования, основных понятиях и подходах при разработке программного обеспечения;

основные понятия объектно-ориентированного программирования;

#### *умения:*

оценка вычислительной сложности разрабатываемых алгоритмов;

разработка моделей информационных систем на основе парадигмы объектно-ориентированного программирования;

применение типовых алгоритмов и структур данных для решения прикладных задач;

разработка прикладных программ для инженерных и научных вычислений (автоматизация обработки наборов данных, визуализация данных, применение численных методов) на языке программирования Python с использованием стандартных и специализированных библиотек;

#### *навыки:*

чтение и понимание исходных кодов программ на языке Python;

разработка программ на языке программирования Python по заданным алгоритмическим схемам;

разработка алгоритмов для решения простых прикладных задач.

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *15.04.03 Прикладная механика*.

Содержание дисциплины является логическим продолжением дисциплин: **ВЫСШАЯ МАТЕМАТИКА В НАУЧНЫХ ИССЛЕДОВАНИЯХ**.

Содержание дисциплины является основой для освоения дисциплин: **МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ПРОЦЕССОВ ОБРАБОТКИ МЕТАЛЛОВ ДАВЛЕНИЕМ, НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ РАБОТА**.

Предварительные компетенции, сформированные у обучающегося до начала изучения дисциплины:

- ОПК-5 — Способен разрабатывать аналитические и численные методы при создании математических моделей машин, приводов, оборудования, систем, технологических процессов

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч.

#### 3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме		Самостоятельная работа студентов	Формируемая компетенция, %
				ВСЕГО	Практические занятия		ОПК-12
5	10	<b>Раздел 1. Технология программирования. Основы алгоритмизации.</b> Технология программирования, основные понятия и подходы. Понятие алгоритма, его свойства, формы представления. Язык программирования Python.	6	2	2	4	10
5	10	<b>Раздел 2. Базовый синтаксис Python.</b> Введение в процедурное программирование. Типы данных. Типы коллекций. Управляющие структуры и функции. Модули. Работа с файлами.	22	8	8	14	25
5	10	<b>Раздел 3. Основные алгоритмические конструкции. Вычислительная сложность.</b> Основные алгоритмические конструкции. Алгоритмы обработки массивов, сортировки и поиска. Вычислительная сложность.	18	8	8	10	15
5	10	<b>Раздел 4. Основы объектно-ориентированного программирования.</b> Основные понятия объектно-ориентированного программирования. Объекты, классы, методы, атрибуты. Инкапсуляция, наследование и полиморфизм.	12	4	4	8	15
5	10	<b>Раздел 5. Стандартные и специализированные библиотеки Python.</b> Стандартные библиотеки Python (math, time, sys, os). Использование интерактивного блокнота Jupyter. Применение библиотеки NumPy для работы с массивами данных. Применение библиотеки Pandas для обработки и анализа данных. Применение библиотеки Matplotlib для визуализации данных. Применение библиотеки SciPy для специализированных инженерных и научных расчетов.	30	10	10	20	20
5	10	<b>Раздел 6. Создание проекта.</b> Создание проекта. Модульность. Системы контроля версий. GIT. Создание графического интерфейса.	20	2	2	18	15
Всего за 10 семестр			108	34	34	74	100
Всего по дисциплине			108	34	34	74	100

#### 3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 1. Технология программирования. Основы алгоритмизации.	Основные концепции и понятия технологии программирования. Интегрированные среды разработки (IDE). Установка и настройка окружения.	2
2	Раздел 2. Базовый синтаксис Python.	Введение в процедурное программирование. Использование интерактивного блокнота Jupyter.	2
3		Управляющие структуры и функции. Модули.	2
4		Работа с файлами.	2
5		Типы данных. Типы коллекций. Модули.	2
6		Разбор основных алгоритмических конструкций.	2
7	Раздел 3. Основные алгоритмические конструкции. Вычислительная сложность.	Алгоритмы обработки массивов, сортировки и поиска.	2
8		Оценка вычислительной сложности.	2
9		Обработка исключений.	2
10	Раздел 4. Основы объектно-ориентированного программирования.	Основные понятия и принципы объектно-ориентированного программирования.	2
11		Разбор примеров применения объектно-ориентированного программирования.	2
12	Раздел 5. Стандартные и специализированные библиотеки Python.	Стандартные библиотеки Python (math, time, sys, os). Использование интерактивного блокнота Jupyter.	2
13		Применение библиотеки Matplotlib для визуализации данных.	2
14		Применение библиотеки NumPy для работы с массивами данных.	2
15		Применение библиотеки Pandas для обработки и анализа данных.	2
16		Применение библиотеки SciPy для специализированных инженерных и научных расчетов.	2
17	Раздел 6. Создание проекта.	Создание простого проекта.	2
Всего за 10 семестр			34

#### 3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Технология программирования. Основы алгоритмизации.	Ознакомление с особенностями наиболее популярных языков программирования. Самостоятельная установка IDE и настройка окружения.	4
2	Раздел 2. Базовый синтаксис Python.	Написание программ использующих базовые структуры данных языка программирования Python с применением простых управляющих структур.	14
3	Раздел 3. Основные алгоритмические конструкции. Вычислительная сложность.	Выполнение домашнего задания на разработку алгоритмов.	2
4		Выполнение задания на обработку массива данных.	3
5		Выполнение задания на сортировку и поиск.	3
6		Выполнение задания на оценку вычислительной сложности.	2
7	Раздел 4. Основы объектно-ориентированного программирования.	Выполнения домашних заданий по реализации объектно-ориентированных моделей в языке программирования Python.	8

8	Раздел 5. Стандартные и специализированные библиотеки Python.	Выполнение домашнего задания с применением библиотеки SciPy.	4
9		Выполнение домашнего задания с применением стандартных библиотек Python.	4
10		Выполнение домашнего задания с применением библиотеки NumPy.	4
11		Выполнение домашнего задания с применением библиотеки Pandas.	4
12		Выполнение домашнего задания с применением библиотеки Matplotlib.	4
13	Раздел 6. Создание проекта.	Выполнение индивидуального практического задания.	18
<b>Всего за 10 семестр</b>			<b>74</b>

#### 4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<b>10</b>						ДР				ДР				ДЗ, ИПЗ		ДР	зач.

Условные обозначения:

- ДР – диагностическая работа;
- ДЗ – домашнее задание;
- ИПЗ – индивидуальное практическое задание;
- зач. – зачет.

**Текущий контроль успеваемости** студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- домашнее задание;
- индивидуальное практическое задание.

**Промежуточная аттестация** проводится в формах:

- зачет.

## 5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 5.1. Основная литература по дисциплине:

1. А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. . Алгоритмизация и программирование. Практикум. Санкт-Петербург: Лань, 2022, эл. рес.
2. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, 450 экз.
3. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, эл. рес.
4. Г. С. Иванова. . Технология программирования. М.: Изд-во МГТУ им. Н. Э. Баумана, 2006, эл. рес.
5. Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python. Москва: Юрайт, 2023, эл. рес.
6. Дж. Кью, М. Джеанини. Объектно-ориентированное программирование. М.: Питер, 2005, 30 экз.
7. П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления. Санкт-Петербург: Питер, 2021, эл. рес.
8. С. З. Свердлов. . Языки программирования и методы трансляции. СПб.: Лань, 2019, 25 экз.
9. Х. М. Дейтел, П. Дж. Дейтел. . Как программировать на C ++. БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2006, эл. рес.

### 5.2. Дополнительная литература по дисциплине:

не требуется.

### 5.3. Периодические издания:

не требуются.

### 5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. <https://www.altlinux.org/Books:PythonSchool> — Books:PythonSchool — ALT Linux Wiki;
2. <https://www.yuripetrov.ru/edu/python/index.html> — Программирование на языке высокого уровня (Python) &mdash; Курс Python (2022);
3. <https://matplotlib.org> — Matplotlib &#8212; Visualization with Python;
4. <https://docs.python.org> — 3.10.6 Documentation;
5. <https://docs.scipy.org> — Numpy and Scipy Documentation &mdash; Numpy and Scipy documentation;
6. <https://numpy.org> — NumPy;
7. <https://pandas.pydata.org> — pandas - Python Data Analysis Library;
8. <https://git-scm.com/book/ru/v2> — Git - Book;
9. <https://docs.python.org/3/library/tkinter.html> — tkinter — Python interface to Tcl/Tk &#8212; Python 3.10.6 documentation;
10. [https://ru.wikibooks.org/wiki/GUI\\_Help/Tkinter\\_book](https://ru.wikibooks.org/wiki/GUI_Help/Tkinter_book) — GUI Help/Tkinter book — Викиучебник.

### Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;  
<http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

### Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;
2. [http://library.voenmeh.ru/jirbis2/index.php?option=com\\_irbis&view=irbis&Itemid=457](http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457) - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/>- КонсультантПлюс- информационный портал правовой информации.

### 5.5. Программное обеспечение:

1. Python 3.4;
2. Spyder;
3. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE.

### 5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.



## **6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **6.1. Практические занятия:**

1. Python 3.4;
2. Spyder;
3. Набор средств трансляции, компоновки, отладки и выполнения Python 3.x с интегрированной средой разработки IDLE.

### **6.2. Прочее:**

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

### Аннотация рабочей программы

Дисциплина **АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *15.04.03 Прикладная механика*. Дисциплина реализуется на факультете *Е Оружие и системы вооружения* БГТУ "ВОЕНМЕХ" им. Д.Ф. Устинова кафедрой *Е4 ВЫСОКОЭНЕРГЕТИЧЕСКИЕ УСТРОЙСТВА АВТОМАТИЧЕСКИХ СИСТЕМ*.

Дисциплина нацелена на формирование *компетенций*:

ОПК-12 Способен создавать алгоритмы цифровой обработки баз данных результатов испытаний и эксплуатации сложных деталей и узлов в машиностроении, разрабатывать современные цифровые программы расчетов и проектирования деталей, узлов, конструкций, машин и материалов с учетом требований надежности, долговечности и безопасности их эксплуатации.

Содержание дисциплины охватывает круг вопросов, связанных с формированием основных представлений об алгоритмизации и программировании, а именно: разработке алгоритмов и оценки их вычислительной сложности, основных парадигм и концепций разработки программного обеспечения, основами объектно-ориентированном программировании. Формирует базовые навыки написания прикладных программ на языке программирования Python, даёт представление об основных возможностях стандартных и специализированных библиотек (NumPy, Pandas, Matplotlib, SciPy), используемых для инженерных и научных вычислений.

Программой дисциплины предусмотрены следующие **виды контроля**:

**Текущий контроль успеваемости** студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- домашнее задание;
- индивидуальное практическое задание.

**Промежуточная аттестация** проводится в формах:

- зачет.

Общая трудоемкость освоения дисциплины составляет **3 з.е., 108 ч.** Программой дисциплины предусмотрены практические занятия (**34 ч.**), самостоятельная работа студента (**74 ч.**).

## ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

### Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 108 ч., из них 34 ч. аудиторных занятий, и 74 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
<b>Раздел 1. Технология программирования. Основы алгоритмизации.</b>		
Ознакомление с особенностями наиболее популярных языков программирования. Самостоятельная установка IDE и настройка окружения.	Г. С. Иванова. . Технология программирования: М.: Изд-во МГТУ им. Н. Э. Баумана, 2006 (1, 2) Х. М. Дейтел, П. Дж. Дейтел. . Как программировать на C ++: БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2006 (1) С. З. Свердлов. . Языки программирования и методы трансляции: СПб.: Лань, 2019 (1) А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1) А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. . Алгоритмизация и программирование. Практикум: Санкт-Петербург: Лань, 2022 (1)	4
Итого по разделу 1		4
<b>Раздел 2. Базовый синтаксис Python.</b>		
Написание программ использующих базовые структуры данных языка программирования Python с применением простых управляющих структур.	П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления: Санкт-Петербург: Питер, 2021 (1-6, 9) Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python: Москва: Юрайт, 2023 (1-11)	14
Итого по разделу 2		14
<b>Раздел 3. Основные алгоритмические конструкции. Вычислительная сложность.</b>		
Выполнение домашнего задания на разработку алгоритмов.	А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1)	2
Выполнение задания на обработку массива данных.	П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления: Санкт-Петербург: Питер, 2021 (3, 4, 7)	3
Выполнение задания на сортировку и поиск.	Т. Кормен, Ч. Лейзерсон, Р. Ривест. . Алгоритмы: построение и анализ: М.: МЦНМО, 2000 (1, 2)	3
Выполнение задания на оценку вычислительной сложности.	А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. . Алгоритмизация и программирование. Практикум: Санкт-Петербург: Лань, 2022 (1-5)	2
Итого по разделу 3		10
<b>Раздел 4. Основы объектно-ориентированного программирования.</b>		
Выполнения домашних заданий по реализации объектно-ориентированных моделей в языке программирования Python.	Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python: Москва: Юрайт, 2023 (12) П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления: Санкт-Петербург: Питер, 2021 (10) Дж. Кью, М. Джеанини. . Объектно-ориентированное программирование: М.: Питер, 2005 (1,2)	8
Итого по разделу 4		8
<b>Раздел 5. Стандартные и специализированные библиотеки Python.</b>		
Выполнение домашнего задания с применением библиотеки SciPy.	П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления: Санкт-Петербург: Питер, 2021 (7)	4
Выполнение домашнего задания с применением стандартных библиотек Python.		4
Выполнение домашнего задания с применением библиотеки NumPy.		4
Выполнение домашнего задания с применением библиотеки Pandas.		4
Выполнение домашнего задания с применением		4

библиотеки Matplotlib.		
Итого по разделу 5		20
<b>Раздел 6. Создание проекта.</b>		
Выполнение индивидуального практического задания.	П. Дж. Дейтел, Х. М. Дейтел. . Python: Искусственный интеллект, большие данные и облачные вычисления: Санкт-Петербург: Питер, 2021 (1 - 10) Д. Ю. Фёдоров. . Программирование на языке высокого уровня Python: Москва: Юрайт, 2023 (14)	18
Итого по разделу 6		18

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- индивидуальное практическое задание;
- домашнее задание;
- зачет.

### Критерии оценивания

#### Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

#### Индивидуальное практическое задание

Индивидуальное практическое задание связано с разработкой программы на языке программирования Python на свободную тему. Индивидуальное задание считается выполненным, если программа выполняет свои функции в соответствии с заданием и студент способен объяснить алгоритм её работы.

#### Домашнее задание

Домашние задания связаны с разработкой программ на языке программирования Python. Тематика программ соответствует темам раздела. Домашнее задание считается выполненным, если программа выполняет свои функции в соответствии с заданием и студент способен объяснить алгоритм её работы.

#### Зачет

Зачёт ставится на основании выполнения домашних заданий и/или индивидуального практического задания в соответствии с технологической картой на дисциплину.

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме		Самостоятельная работа студентов	Формируемая компетенция, %	НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Практические занятия		ОПК-12	
5	10	Раздел 1. Технология программирования. Основы алгоритмизации.	6	2	2	4	10	Индивидуальное практическое задание
5	10	Раздел 2. Базовый синтаксис Python.	22	8	8	14	25	Домашнее задание
5	10	Раздел 3. Основные алгоритмические конструкции. Вычислительная сложность.	18	8	8	10	15	Индивидуальное практическое задание
5	10	Раздел 4. Основы объектно-ориентированного программирования.	12	4	4	8	15	Домашнее задание
5	10	Раздел 5. Стандартные и специализированные библиотеки Python.	30	10	10	20	20	Индивидуальное практическое задание
5	10	Раздел 6. Создание проекта.	20	2	2	18	15	Индивидуальное практическое задание
Всего за 10 семестр			108	34	34	74	100	
Всего по дисциплине			108	34	34	74	100	

**ОПК-12 - Способен создавать алгоритмы цифровой обработки баз данных результатов испытаний и эксплуатации сложных деталей и узлов в машиностроении, разрабатывать современные цифровые программы расчетов и проектирования деталей, узлов, конструкций, машин и материалов с учетом требований надежности, долговечности и безопасности их эксплуатации**

№ 1 Прочитайте текст и запишите развернутый обоснованный ответ

Есть список студентов с полученными баллами. Нужно написать программу, переводящую полученные баллы в оценку.

Например:

90 - 100 баллов → оценка «отлично»

75 - 89 баллов → оценка «хорошо»

60 - 74 баллов → оценка «удовлетворительно»

менее 60 баллов → оценка «неудовлетворительно»

Какие основные алгоритмические конструкции Вы будете использовать при написании программы? Опишите их последовательность

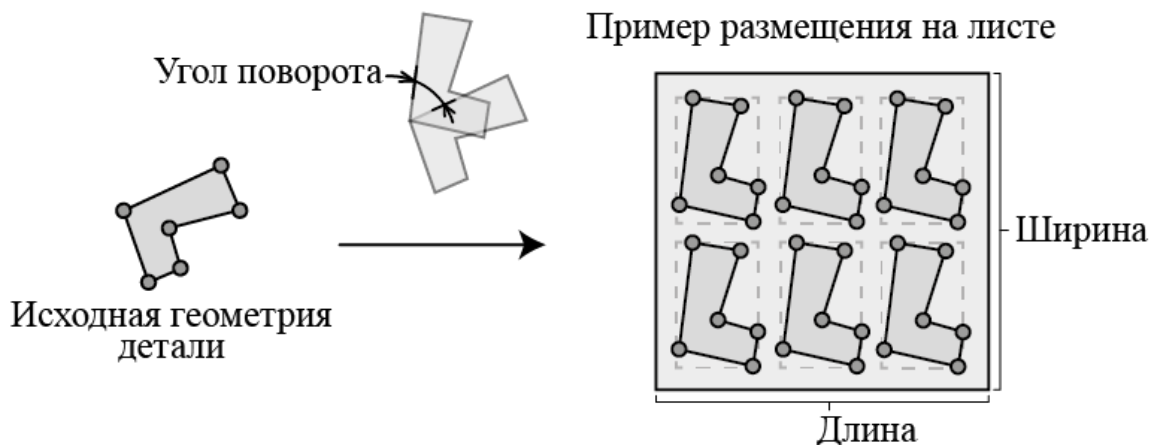
№ 2 Прочитайте текст и запишите развернутый обоснованный ответ

Предложите простой алгоритм для оптимизации раскроя металлического листа под вырубку детали произвольной формы. Параметром оптимизации является угол поворота детали.

Входные данные:

- геометрия детали (массив точек контура);
- габариты листа (ширина, длина);
- минимальная ширина перемычки;
- допустимые углы поворота (например, с шагом 15°).

Для простоты задачи можно принять, что вид раскроя – прямой.



№ 3 Прочитайте текст и установите последовательность

Составьте из следующих строк программу, вычисляющую сумму цифр числа (язык программирования - Python)

1. `n = input('Введите число: ')`

2. `total = 0`

3. `for digit in n:`

4. `total += int(digit)`

5. `print(f'Сумма цифр: {total}')`

№ 4 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа

Во сколько примерно увеличится время выполнения алгоритма с вычислительной сложностью  $O(n^2)$ , если размерность входных данных  $n$  увеличить в два раза?

1. Не изменится

2. Увеличится в два раза
3. Увеличится в четыре раза
4. Уменьшится в четыре раза

№ 5 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа  
Что описывает нотация O-большое?

1. Точное время выполнения алгоритма
2. Верхнюю границу роста времени выполнения алгоритма при увеличении размерности входных данных
3. Нижнюю границу сложности алгоритма
4. Средний случай производительности алгоритма

№ 6 Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа  
К какому типу языков программирования относится язык программирования Python

1. Язык с динамической сильной (строгой) неявной типизацией
2. Язык со статической сильной (строгой) неявной типизацией
3. Язык с динамической слабой (не строгой) явной типизацией
4. Язык со статической слабой (не строгой) явной типизацией

№ 7 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов  
Какие из строчек программы на языке программирования Python содержат ошибки?

1. `print("Hello world!")`
2. `if x == 5 print(x)`
3. `for i in range(5): print(i)`
4. `def func(): return`

№ 8 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов  
Выберите выражения, являющиеся зарезервированными ключевыми словами языка программирования Python

1. `def`
2. `True`
3. `solve`
4. `check`

№ 9 Прочитайте текст, выберите правильные ответы и запишите аргументы, обосновывающие выбор ответов  
Какие операции вернут целочисленный результат? (язык программирования Python 3)

1. `10 + 3`
2. `10 / 3`
3. `10 // 3`
4. `10. + 3`

№ 10 Прочитайте текст и установите соответствие

Соотнесите значения переменных и соответствующие им типы данных (язык программирования Python)

- |                          |                                                   |
|--------------------------|---------------------------------------------------|
| 1. <code>a = 1</code>    | A. <code>int</code> (целые числа)                 |
| 2. <code>a = 1.1</code>  | Б. <code>float</code> (числа с плавающей запятой) |
| 3. <code>a = '1'</code>  | В. <code>str</code> (строковые числа)             |
| 4. <code>a = True</code> | Г. <code>boolen</code> (логический тип)           |
|                          | Д. <code>tuple</code> (кортеж)                    |

№ 11 Прочитайте текст и установите соответствие

Установите соответствие к какому типу данных относится принимаемое переменной значение (язык программирования Python)

- |                                                      |                                             |
|------------------------------------------------------|---------------------------------------------|
| 1. <code>a = [1, 2, 3]</code>                        | A. Изменяемый тип данных (mutable)          |
| 2. <code>a = (1, 2, 3)</code>                        | Б. Неизменяемый тип данных (immutable)      |
| 3. <code>a = {1: 'one', 2: 'two', 3: 'three'}</code> | В. Взаимозаменяемый тип данных (permutable) |
| 4. <code>a = "123.43"</code>                         |                                             |
| 5. <code>a = {1, 2, 3}</code>                        |                                             |

№ 12 Прочитайте текст и установите последовательность

Есть пять разных алгоритмов, выполняющих одну и ту же задачу. Расположите их в порядке возрастания времени их выполнения при увеличении размера входных данных  $n$ :

1.  $O(\log n)$
2.  $O(n)$
3.  $O(n \cdot \log n)$



4.  $O(n^2)$

5.  $O(n!)$