

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
НИЖЕГОРОДСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное  
учреждение «НИЖЕГОРОДСКИЙ КОЛЛЕДЖ МАЛОГО БИЗНЕСА»  
Центр цифрового образования детей «IT-куб»**

**СОГЛАСОВАНО**

**Руководитель ЦЦОД «IT-куб»**

**А.А. Полякова**

**«19» августа 2025 год**

**УТВЕРЖДАЮ**

**Директор ГБПОУ НКМБ**

**А.С. Евтеев**

**«19» августа 2025 год**

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА  
«Программирование на языке Python»  
*Направленность – техническая***

**Возраст обучающихся: 12-16 лет**

**Объем: 148 часов**

**Автор-составитель:**

**Бельский Александр Сергеевич,  
педагог дополнительного образования**

**Нижний Новгород**

**2025**

## СОДЕРЖАНИЕ

1. Комплекс основных характеристик программы	3
1.1 Пояснительная записка	3
1.2 Цель и задачи программы	5
1.3 Содержание общеразвивающей программы	6
1.3.1 Учебный план	6
1.3.2 Содержание учебного плана	8
1.4 Требования к результатам освоения программы	11
2. Комплекс организационно-педагогических условий реализации общеразвивающей программы	12
2.1 Календарный учебный график на 2025-2026 учебный год	12
2.2 Условия реализации программы	13
2.3 Формы аттестации и оценочные материалы	14
2.4 Методические материалы	15
Список литературы	17

# 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ

## 1.1 Пояснительная записка

**Направленность и уровень программы.** Программа «Программирование на языке Python» имеет техническую направленность. Уровень - базовый.

**Актуальность программы.** Программа позволяет школьникам освоить ключевые навыки программирования, алгоритмического и системного мышления. Эти знания развивают понимание построения алгоритмов, понимание «Интернета вещей», принципа действия разных устройств, техническую грамотность и подготовят учащихся к современным профессиональным вызовам.

Основанием для проектирования и реализации данной общеразвивающей программы служит *перечень следующих нормативных правовых актов и государственных программных документов:*

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 г. № 273-ФЗ;
- Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;
- Письмо Министерства образования и науки РФ от 18.11.2015 г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение правительства РФ от 04.09.2014 г. № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- «Основы законодательства РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 г. № 5487 - (ред. от 25.11.2009 г.);
- Федеральный закон от 24.07.1998 г. № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»;
- Федеральный закон от «Об основах охраны здоровья граждан в

Российской Федерации», 2011 г.;

– Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен).

**Адресат программы.** Программа предназначена для детей относящихся к возрастной группе 12-16 лет.

**Форма обучения** – очная, с возможностью применения дистанционных технологий. (Закон №273-ФЗ, гл.2, ст.17, п.2.).

**Режим занятий.** Занятия проводятся 2 раза в неделю по два занятия. Продолжительность одного занятия - 45 минут, так как обучение проходит с использованием компьютерной техники. После 45 минут занятия организовывается обязательный перерыв 10 минут.

**Срок реализации программы.** 9 месяцев.

**Объём программы.** 148 часов.

**Формы занятий.** Групповые, количество обучающихся в группе – 6-12 человек.

**Место проведения занятий:** 603136, Нижегородская область, г. Нижний Новгород, ул. Маршала Малиновского, д. 1.

### ***Аннотация***

Программа «Программирование на языке Python» имеет техническую и творческую направленность, в ходе обучения учащиеся знакомятся с основами программирования и алгоритмов, развивают креативное, техническое и критическое мышление.

Программа позволяет учащимся получить необходимый объем знаний в зависимости от уровня подготовки и интересов. Программа рассчитана на обучающихся 12-16 лет.

## 1.2 Цель и задачи программы

**Цель:** обучение языку программирования Python.

**Задачи:**

*Образовательные:*

1. Познакомить с конструкциями языка программирования Python;
2. Познакомить с основными структурами данных;
3. Сформировать навыки работы в интегрированной среде разработки на языке Python;
4. Сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
5. Сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ.

*Развивающие:*

1. Способствовать развитию навыков проектной деятельности;
2. Способствовать развитию навыков поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
3. Сформировать у обучающихся интерес к программированию, самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники.

*Воспитательные:*

1. Способствовать воспитанию этики групповой работы;
2. Сформировать коммуникативную культуру обучающихся, как внутри проектных групп, так и в коллективе в целом.

## 1.3 Содержание общеразвивающей программы

### 1.3.1 Учебный план

Таблица 1

№ п/п	Название темы	Количество часов		
		Всего	Теория	Практика
<b>Раздел 1. Введение в программирование</b>		<b>6</b>	<b>3</b>	<b>3</b>
1.1.	Знакомство со средой программирования Python.	2	1	1
1.2.	Переменные и операторы языка программирования Python.	4	2	2
<b>Раздел 2. Условный оператор и циклы в языке программирования Python</b>		<b>52</b>	<b>26</b>	<b>26</b>
2.1.	Конструкция if.	4	2	2
2.2.	Конструкции elif, else.	4	2	2
2.3.	Объединение условий: применение условных конструкций на практике.	4	2	2
2.4.	Область видимости переменных, Создание функций.	4	2	2
2.5.	Изучение цикла for.	4	2	2
2.6.	Изучение функции range.	4	2	2
2.7.	Использование цикла for для решения базовых задач.	4	2	2
2.8.	Изучение цикла while.	4	2	2
2.9.	Изучение вложенных циклов, инструкций break, continue.	4	2	2
2.10.	Использование цикла while для решения базовых задач.	4	2	2
2.11.	Применение цикла for для решения сложных задач.	4	2	2
2.12.	Применение цикла while для решения сложных задач.	4	2	2
2.13.	Применение условных конструкций для решения сложных задач.	4	2	2
<b>Раздел 3. Изучение строк, списков, кортежей, методов и словарей в языке программирования Python</b>		<b>80</b>	<b>35</b>	<b>45</b>
3.1.	Операции со строками.	4	2	2
3.2.	Использование строк для решения базовых задач.	2	1	1
3.3.	Подготовка к промежуточному контролю.	2	1	1
3.4.	Промежуточный контроль.	2	0	2
3.5.	Использование строк для решения базовых задач.	2	1	1
3.6.	Изучение некоторых методов у строк.	4	2	2
3.7.	Использование методов при решении базовых задач.	4	2	2
3.8.	Изучение управляющих символов и извлечение среза.	4	2	2

3.9.	Применение методов строк для решения сложных задач.	4	2	2
3.10.	Изучение списков.	4	2	2
3.11.	Использование методов при решении базовых и сложных задач.	4	2	2
3.12.	Изучение массивов.	4	2	2
3.13.	Использование массивов при решении базовых и сложных задач.	4	2	2
3.14.	Изучение кортежей.	4	2	2
3.15.	Использование кортежей при решении задач.	4	2	2
3.16.	Изучение методов.	4	2	2
3.17.	Использование методов при решении базовых задач.	4	2	2
3.18.	Изучение словарей.	4	2	2
3.19.	Использование словарей при решении базовых задач.	4	2	2
3.20.	Использование строк, списков, методов, кортежей и списков для решения сложных задач.	4	2	2
3.21.	Практические занятия, решение задач.	4	0	4
3.22.	Практические занятия, решение задач.	4	0	4
<b>Раздел 4. Практикум</b>		<b>6</b>	<b>3</b>	<b>3</b>
4.1.	Изучение структуры создания проекта и выбор темы.	2	1	1
4.2.	Составление кода.	2	1	1
4.3.	Предварительный показ кода и презентации проекта.	2	1	1
<b>Раздел 5. Презентация проекта</b>		<b>2</b>	<b>1</b>	<b>1</b>
5.1.	Презентация проекта.	2	1	1
<b>Итоговая аттестация</b>		<b>2</b>	<b>0</b>	<b>2</b>
<b>Итого</b>		<b>148</b>	<b>68</b>	<b>80</b>

### 1.3.2 Содержание учебного плана

Таблица 2

Наименование разделов и тем	Краткое содержание темы	Кол-во часов	
		Теория	Практика
<b>Раздел 1. Введение в программирование</b>		<b>3</b>	<b>3</b>
1.1. Знакомство со средой программирования PyCharm.	Проведение ознакомительной беседы с учениками для определения уровня компетенции, познакомится со средой программирования Python.	1	1
1.2. Переменные и операторы языка программирования Python.	Изучение переменных, правил их написания, изучение операторов, разбор задач, написание несложных программ.	2	2
<b>Раздел 2. Условный оператор и циклы в языке программирование Python</b>		<b>26</b>	<b>26</b>
2.1. Конструкция if.	Изучим конструкцию if, принципы работы.	2	2
2.2. Конструкции elif, else.	Изучение конструкций elif, else, if-then-else. Написание кода, используя изученные конструкции.	2	2
2.3. Объединение условий: применение условных конструкций на практике.	Научиться применять конструкции elif, else, if-then-else, if на практике.	2	2
2.4. Область видимости переменных, Создание функций.	Изучение областей видимости переменных, локальная или глобальная, написание простых функций.	2	2
2.5. Изучение цикла for.	Ознакомление с циклом for. Разбор задач. Решение задач с применением цикла for.	2	2
2.6. Изучение функции range.	Изучение функции range. Разбор задач. Решение задач с применением функции range.	2	2
2.7. Использование цикла for для решения базовых задач.	Применение на практических заданиях циклов for.	2	2
2.8. Изучение цикла while.	Изучение цикла while. Решение задач с применением цикла while.	2	2
2.9. Изучение вложенных циклов, инструкций break, continue.	Изучение вложенных циклов. Знакомство с конструкциями break, continue. Решение задач с вложенными циклами. Решение задач с использованием конструкций break, continue.	2	2
2.10. Использование цикла while для решения базовых задач.	Научиться использовать цикл while при решении базовых задач.	2	2

2.11. Применение цикла for для решения сложных задач.	Применение цикла for в практических заданиях.	2	2
2.12. Применение цикла while для решения сложных задач.	Разбор и решение сложных задач с применением цикла while.	2	2
2.13. Применение условных конструкций для решения сложных задач.	Разбор и решение сложных задач с использованием условных конструкций.	2	2
<b>Раздел 3. Изучение строк, списков, кортежей, методов и словарей в языке программирования Python</b>		<b>35</b>	<b>45</b>
3.1. Операции со строками.	Изучение строк и операций над ними. Решение задач.	2	2
3.2. Использование строк для решения базовых задач.	Разбор и решение базовых задач с использованием строк.	1	1
3.3. Подготовка к промежуточному контролю.	Обсуждение общих вопросов.	1	1
3.4. Промежуточный контроль.	Прохождение промежуточного тестирования.	0	2
3.5. Использование строк для решения базовых задач.	Разбор и решение базовых задач с использованием строк.	1	1
3.6. Изучение некоторых методов у строк.	Изучение понятия метод. Ознакомление с разновидностью методов. Решение и разбор задач.	2	2
3.7. Использование методов при решении базовых задач.	Разбор и решение базовых задач с применением методов.	2	2
3.8. Изучение управляющих символов и извлечение среза.	Ознакомление с понятием срез. Изучение управляющих символов и их разновидности.	2	2
3.9. Применение методов строк для решения сложных задач.	Разбор и решение сложных задач с применением методов.	2	2
3.10. Изучение списков.	Изучение понятия список, изучение индексации.	2	2
3.11. Использование методов при решении базовых и сложных задач.	Разбор и решение сложных задач с применением методов.	2	2
3.12. Изучение массивов.	Ознакомление с понятием массива, индексации.	2	2
3.13. Использование массивов при решении	Научимся использовать массивы в решении сложных задач.	2	2

базовых и сложных задач.			
3.14. Изучение кортежей.	Узнаем понятие кортеж, его свойства.	2	2
3.15. Использование кортежей при решении задач.	Применение кортежей для решения практических задач.	2	2
3.16. Изучение методов.	Разбор основных методов. Разбор и решение задач.	2	2
3.17. Использование методов при решении базовых задач.	Научиться использовать методы в решении базовых задач.	2	2
3.18. Изучение словарей.	Ознакомление с понятием словарь. Изучение разновидности словарей.	2	2
3.19. Использование словарей при решении базовых задач.	Научиться использовать словари в решении базовых задач.	2	2
3.20. Использование строк, списков, методов, кортежей и списков для решения сложных задач.	Практическое применение полученных знаний.	2	2
3.21. Практические занятия, решение задач.	Практические занятия, решение задач.	0	4
3.22. Практические занятия, решение задач	Практические занятия, решение задач.	0	4
<b>Раздел 4. Практикум</b>		<b>3</b>	<b>3</b>
4.1. Изучение структуры создания проекта и выбор темы.	Ознакомление со структурой создания проекта и выбор темы.	1	1
4.2. Составление кода.	Разработка алгоритма написания кода. Выделение необходимых для написания кода тем для их более детального изучения.	1	1
4.3. Предварительный показ кода и презентации проекта.	Устранение ошибок в презентации проекта.	1	1
<b>Раздел 5. Презентация проекта</b>		<b>1</b>	<b>1</b>
5.1. Презентация проекта	Презентовать созданный проект комиссии.	1	1
<b>Итоговая аттестация.</b>	Написание итогового тестирования для фиксирования прогресса обучающихся.	<b>0</b>	<b>2</b>
<b>Итого часов: 148</b>		<b>68</b>	<b>80</b>

## 1.4 Требования к результатам освоения программы

### *Предметные результаты:*

1. Ознакомлены с конструкциями языка программирования Python;
2. Ознакомлены с основными структурами данных;
3. Сформированы навыки работы в интегрированной среде разработки на языке Python;
4. Сформированы навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
5. Сформированы и развиты навыки алгоритмического и логического мышления, грамотной разработки программ.

### *Личностные результаты:*

1. Развиты навыки проектной деятельности;
2. Развиты навыки поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
3. Сформирован у обучающихся интерес к программированию, самостоятельность и творческий подход к решению задач с использованием средств вычислительной техники.

### *Метапредметные результаты:*

1. Сформирован навык этики групповой работы;
2. Сформирован навык коммуникативной культуры обучающихся, как внутрипроектных групп, так и в коллективе в целом.

## 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ РЕАЛИЗАЦИИ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

### 2.1 Календарный учебный график на 2025-2026 учебный год

Таблица 3

Месяц	сентябрь					октябрь				ноябрь				декабрь				январь				февраль				март				апрель				май						
Даты	01.09-05.09	08.09-12.09	15.09-19.09	22.09-26.09	29.09-03.10	06.10-10.10	13.10-17.10	20.10-24.10	27.10-31.10	03.11-07.11	10.11-14.11	17.11-21.11	24.11-28.11	01.12-05.12	08.12-12.12	15.12-19.12	22.12-26.12	29.12-02.01	05.01-09.01	12.01-16.01	19.01-23.01	26.01-30.02	02.02-06.02	09.02-13.02	16.02-20.02	23.02-27.02	02.03-06.03	09.03-13.03	16.03-20.03	23.03-27.03	30.03-03.04	06.04-10.04	13.04-17.04	20.04-24.04	27.04-01.05	04.05-08.05	11.05-15.05	18.05-22.05	25.05-29.05	
недели	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	
часы	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2			4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	2

Условные обозначения:

	Занятия по расписанию
	Каникулярный период
	Промежуточная и итоговая аттестация

## 2.2 Условия реализации программы

*Материально-техническое обеспечение:*

1. Ноутбуки – 13 шт.;
2. Интерактивная доска – 1 шт.;
3. Наушники – 6 шт.;
4. Компьютерная мышь – 13 шт.

*Информационное обеспечение:*

1. Пакет библиотек со SciPy: numpy, scipy, matplotlib, ipython + ipythonnotebook, sympy, panda.

### 2.3 Формы аттестации и оценочные материалы

Система отслеживания результатов, обучающихся выстроена следующим образом:

- текущий контроль;
- промежуточный контроль;
- итоговая аттестация.

Входного контроля при приёме по данной общеразвивающей программе не предусмотрено.

Текущий контроль осуществляется путём наблюдения, опроса.

Промежуточный контроль осуществляется в форме прохождения тестирования. Критерии оценивания и оценочные материалы находятся в Приложении.

Итоговая аттестация обучающихся осуществляется в форме итогового тестирования и оценивается по 100-балльной шкале, которая переводится в один из уровней освоения образовательной программы согласно таблице 4. Критерии оценивания и оценочные материалы находятся в Приложении.

Таблица 4

<b>Баллы</b>	<b>Процент освоения программы</b>	<b>Уровень освоения</b>
0-30	0-30%	Низкий
31-70	31-70%	Средний
71-100	71-100%	Высокий

## 2.4 Методические материалы

В рамках реализации программы применяются следующие методы обучения:

- словесный: рассказ, беседа;
- практический: показ, выполнение практических работ и т.д.;
- объяснительно-иллюстративный: рассказ, показ, фильм и т.п.;
- репродуктивный: воспроизведение, действие по алгоритму;
- эвристический: частично-поисковый, самостоятельное нахождение

ответов на поставленные педагогом вопросы;

- проблемный: постановка проблемных вопросов, создание проблемных ситуаций;

- проектный метод: разработка проектов, создание творческих работ.

Большую часть при реализации образовательной деятельности занимают активные и интерактивные методы в соответствии с возрастными особенностями обучающихся.

При реализации программы используются следующие педагогические технологии: группового обучения, специальные технологии, соответствующие технической направленности; коллективного взаимообучения, дифференцированного обучения, проблемного обучения. Особое внимание уделяется использованию в учебном процессе здоровьесберегающих технологий, способствующих предотвращению состояний переутомления, гиподинамии (физминутки, зарядки для глаз и т.д.). Используются следующие формы занятий: комбинированное занятие, практикум, урок-презентация, мастер-класс, конкурс, соревнование, игра и т.д. По дидактической цели занятия делятся на вводные, занятия по углублению знаний, практические занятия, занятия по систематизации и обобщению знаний, по контролю знаний, умений и навыков.

Структура учебного занятия строится в рамках технологии развития критического мышления и включает следующие этапы: вызов (мотивация к изучению материала), осмысление (изучение, повторение, закрепление учебного

материала), рефлексия (подведение итогов, рефлексия эмоционального состояния, саморефлексия и т.д).

## СПИСОК ЛИТЕРАТУРЫ

### *Нормативно-правовые акты:*

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 г. № 273-ФЗ;
- Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р;
- Письмо Министерства образования и науки РФ от 18.11.2015 г. № 09-3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)»;
- Распоряжение правительства РФ от 04.09.2014 г. № 1726-р «Об утверждении Концепции развития дополнительного образования детей»;
- «Основы законодательства РФ об охране здоровья граждан», утвержденные Верховным советом РФ от 22.07.1993 г. № 5487 - (ред. от 25.11.2009 г.);
- Федеральный закон от 24.07.1998 г. № 124-ФЗ «Об основных гарантиях прав ребенка в РФ»;
- Федеральный закон «Об основах охраны здоровья граждан в Российской Федерации», 2011 г.;
- Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам» (Приказ №1008 отменен);
- Санитарно-эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14.

### *Учебная литература:*

- Задачи по программированию. Под ред. С. М. Окулова, М.: БИНОМ. Лаборатория знаний, 2006 г.;
- Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И. Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014 г.;

- Окулов С. М. Основы программирования. М.: Бином. Лаборатория знаний, 2012 г.;
- Поляков К. Ю., Еремин Е. А. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014 г.;
- Эльконин, Д. Б. Детская психология: учеб. пособие для студ. высш. учеб. заведений / Д. Б. Эльконин; ред.сост. Б. Д. Эльконин. – 4-е изд., стер. – М.: Издательский центр «Академия», 2007 г. – 384 с.;
- М. Лутц. Изучаем Python. СПб.: Символ-Плюс, 2011 г.

## Оценочные материалы для промежуточного контроля

### Тестирование

1. Обязательно ли в Пайтоне объявлять тип переменной?

Варианты ответов:

- а) по желанию программиста
- б) обязательно
- в) зависит от ситуации
- г) объявлять тип не нужно

2. Какие из предлагаемых имен допустимы в Python?

Варианты ответов:

- а) s\_5
- б) \_5s
- в) 5\_s
- г) Щ5.

3. Чем отличаются операторы = и ==?

Варианты ответов:

- а) операторы эквивалентны
- б) оператор = менее точный
- в) оператор = присваивает значения, а == сравнивает их
- г) оператор == в Питоне не используется

4. Что будет выведено на экран в результате выполнения инструкции `print(4 + 3.0)`?

Варианты ответов:

- а) 7.0
- б) 4 + 3.0
- в) 7
- г) ошибка

5. Что будет выведено на экран в результате выполнения кода условия?

Варианты ответов:

а) 7 + 3

б) 10

в) 73

г) ошибка

6. Что возвращает функция `len()` при передаче в неё строки?

Варианты ответов:

а) количество слов

б) количество символов

в) количество байт

г) ошибка

7. Что будет выведено на экран в результате выполнения кода условия?

Варианты ответов:

а) 'голливуд'

б) 'холливуд'

в) ничего

г) ошибка

8. Какие объекты можно хранить в списках?

Варианты ответов:

а) только числа и строки

б) только неизменяемые объекты

в) любые, кроме объектов функций

г) любые

9. Что будет выведено на экран в результате выполнения кода условия?

```
li = [1, 2, 3]
```

```
li.extend('abc')
```

```
print(li)
```

Варианты ответов:

а) [1, 2, 3]

б) [1, 2, 3, 'abc']

в) [1, 2, 3, 'a', 'b', 'c']

г) ошибка

10. Как в Питоне обратить порядок элементов в списке?

Варианты ответов:

а) `li.reverse()`

б) `li = li[::-1]`

в) `reverse(li)`

г) никак

**Критерии оценивания промежуточного контроля**

Промежуточное тестирование содержит 10 вопросов по пройденному материалу, критерии оценки – по 10 баллов за каждый правильный ответ, всего 100 баллов.

**Оценочные материалы для итоговой аттестации****Тестирование**

1. Какие из представленных символов или их комбинаций используются в Python для комментирования кода?

Варианты ответов:

- а) #
- б) //
- в) <!-- -->
- г) /\* \*/

2. Для выделения блоков кода одного уровня вложенности в Питоне используются...

Варианты ответов:

- а) круглые скобки
- б) квадратные скобки
- в) фигурные скобки
- г) идентичные отступы

3. Как в Питоне записать выражение а не равно b?

Варианты ответов:

- а)  $a < > b$
- б)  $a != b$
- в)  $a \text{ is not } b$
- г)  $a \neq b$

4. Какой тип данных в Python представляет целые числа?

Варианты ответов:

- а) complex
- б) float
- в) int
- г) bool

5. Какое число будет выведено на экран в результате выполнения кода условия?

```
x = 5
```

```
y = 2
```

```
z = x%y
```

```
print(z)
```

Варианты ответов:

а) 2.5

б) 1

в) 0.1

г) ошибка

6. Что будет выведено на экран в результате выполнения инструкции `print("123456789"[5])`?

Варианты ответов:

а) 12345

б) 5

в) 6

г) ошибка

7. Какой из методов используется для преобразования строки в верхний регистр?

Варианты ответов:

а) `upper()`

б) `capitalize()`

в) `lower()`

г) `title()`

8. Какие из представленных литералов относятся к типу `list`?

Варианты ответов:

а) `['1', '2', '3']`

б) `'[1, 2, 3]'`

в) `[(1, 2, 3)]`

г) ([1], [2], [3])

9. Выберите верные утверждения о кортежах.

Варианты ответов:

а) неизменяемые

б) изменяемые

в) неупорядоченные

г) упорядоченные

10. Как удалить из кортежа `tpl` элемент `elem` с индексом `i`?

Варианты ответов:

а) `tpl.pop(i)`

б) `del tpl[i]`

в) `tpl.remove(elem)`

г) никак

11. Какой тип данных в Python представляет словари?

Варианты ответов:

а) `dict`

б) `list`

в) `tuple`

г) `set`

12. Какие из утверждений о словарях верны?

Варианты ответов:

а) изменяемые

б) неизменяемые

в) доступ к элементам по индексу

г) доступ к элементам по ключу

13. Что будет выведено на экран в результате выполнения кода условия?

```
d = {1: '1'}
```

```
d += d*2
```

```
print(d)
```

Варианты ответов:

- a) {1, 1, 1}
- б) {'1', '1', '1'}
- в) {1: '1', 1: '1', 1: '1'}
- г) ошибка

14. Какой тип данных в Питоне представляет множества?

Варианты ответов: а) tuple, б) list, в) dict, г) set.

15. Что будет выведено на экран в результате выполнения кода условия?

```
st = set()
st.add(1)
st.add((2, 3))
print(st)
```

Варианты ответов: а) {1, 2, 3}, б) {(2, 3), 1}, в) {2, 3}, г) ошибка.

16. Какой оператор позволяет проверить одновременное выполнение сразу нескольких условий?

Варианты ответов: а) and, б) or, в) not, г) is.

17. Что будет выведено на экран в результате выполнения кода условия?

Варианты ответов: а) 10, б) 20, в) 30, г) ошибка.

18. Что будет выведено на экран в результате выполнения кода условия?

a, b = 5, 10

if a>b:

    print("a>b")

else:

    print("a<=b")

Варианты ответов: а) a > b, б) a >= b, в) a < b, г) a <= b.

19. Что будет выведено на экран в результате выполнения кода условия?

for i in range(3):

    print(-i, end=' ')

Варианты ответов: а) -1 -2 -3, б) 0 -1 -2 -3, в) -1 -2 -3, г) 0 -1 -2.

20. Что будет выведено на экран в результате выполнения кода условия?

k = 1

```
while k <= 5:  
    if k > 3: break  
    print(k, end=' ')  
    k += 1
```

Варианты ответов: а) 1 2 3, б) 1 2 3 4, в) 1 2 3 4 5, г) ошибка.

**Критерии оценивания итогового тестирования**

Итоговое тестирование содержит 20 вопросов по пройденному материалу, критерии оценки – по 5 баллов за каждый правильный ответ, всего 100 баллов.