

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«КАРАЧАЕВО-ЧЕРКЕССКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИМЕНИ У.Д. АЛИЕВА»**

Факультет экономики и управления

УТВЕРЖДАЮ

И. о. проректора по УР

М. Х. Чанкаев

«29» апреля 2026 г., протокол № 7

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

ЯЗЫКИ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ

(наименование дисциплины (модуля))

Направление подготовки

09.03.03 – Прикладная информатика

(шифр, название направления)

Квалификация выпускника

Бакалавр

ТЕСТОВЫЕ ВОПРОСЫ ПО ДИСЦИПЛИНЕ «ЯЗЫКИ ПРОГРАММИРОВАНИЯ ВЫСОКОГО УРОВНЯ»

ПК-2

1. Задания закрытого типа с одним правильным ответом (5 заданий)

1.1. Какая команда используется для вывода данных на экран в Python?

- А) input()
- Б) print()
- В) output()
- Г) write()

1.2. Какой оператор используется для проверки условия в Python?

- А) for
- Б) while
- В) if
- Г) def

1.3. Какой результат выполнения выражения `int("25") + 5`?

- А) "255"
- Б) 30
- В) 25
- Г) Ошибка

1.4. Какое ключевое слово используется для определения функции?

- А) function
- Б) def
- В) fun
- Г) define

1.5. Какой метод строки возвращает её длину?

- А) size()
- Б) length()
- В) len()
- Г) count()

2. Задания закрытого типа с несколькими правильными ответами (5 заданий)

2.1. Выберите все корректные имена переменных в Python:

- А) _name
- Б) 2var
- В) myVar
- Г) class
- Д) user_name

2.2. Какие из перечисленных типов данных являются изменяемыми (mutable)?

- А) list
- Б) tuple
- В) dict
- Г) str
- Д) set

2.3. Какие операторы используются для организации циклов в Python?

- А) for
- Б) while
- В) do-while
- Г) repeat
- Д) loop

2.4. Какие из перечисленных методов относятся к спискам?

- А) append()
- Б) upper()

- В) extend()
- Г) split()
- Д) pop()

2.5. Какие утверждения о рекурсии верны?

- А) Рекурсивная функция вызывает саму себя
- Б) Рекурсия всегда быстрее цикла
- В) Для рекурсии нужен базовый случай
- Г) Рекурсия невозможна в Python
- Д) Отсутствие базового случая приводит к переполнению стека

3. Задания закрытого типа. Задачи на соответствие (5 заданий)

3.1. Установите соответствие между оператором и его назначением:

Оператор	Назначение
1) print()	А) Ввод данных
2) input()	Б) Вывод данных
3) int()	В) Преобразование к целому
4) len()	Г) Длина объекта

3.2. Установите соответствие между типом данных и примером:

Тип	Пример
1) int	А) "Hello"
2) str	Б) [1, 2, 3]
3) list	В) 42
4) dict	Г) {"a": 1}

3.3. Установите соответствие между циклом и его особенностью:

Цикл	Особенность
1) for	А) Выполняется, пока условие истинно
2) while	Б) Перебирает элементы последовательности
3) break	В) Прерывает цикл
4) continue	Г) Переходит к следующей итерации

3.4. Установите соответствие между методом строки и его действием:

Метод	Действие
1) upper()	А) Разбивает строку на список
2) split()	Б) Возвращает строку в верхнем регистре

Метод	Действие
3) replace()	В) Заменяет подстроку
4) strip()	Г) Удаляет пробелы по краям

3.5. Установите соответствие между понятием ООП и его описанием:

Понятие	Описание
1) Класс	А) Экземпляр класса
2) Объект	Б) Шаблон для создания объектов
3) Метод	В) Функция внутри класса
4) __init__	Г) Конструктор класса

4. Задания закрытого типа на установление правильной последовательности (5 заданий)

4.1. Установите правильную последовательность установки интерпретатора Python:

1. Скачать установочный файл с официального сайта
2. Запустить установщик
3. Отметить пункт «Add Python to PATH»
4. Дождаться завершения установки
5. Проверить установку командой `python --version`

4.2. Установите правильную последовательность выполнения условного оператора:

1. Вычисление условия
2. Если условие истинно — выполнение блока `if`
3. Если условие ложно — переход к `elif` или `else`
4. Выполнение соответствующего блока
5. Продолжение выполнения программы

4.3. Установите правильную последовательность работы цикла `for`:

1. Получение следующего элемента последовательности
2. Присваивание элемента переменной цикла
3. Выполнение тела цикла
4. Проверка наличия следующего элемента
5. Завершение цикла

4.4. Установите правильную последовательность создания и использования функции:

1. Определение функции через `def`
2. Указание параметров функции
3. Написание тела функции
4. Возврат значения через `return`
5. Вызов функции с аргументами

4.5. Установите правильную последовательность создания пользовательского класса:

1. Объявление класса через `class`
2. Определение конструктора `__init__`
3. Определение методов класса
4. Создание экземпляра класса
5. Вызов методов объекта

5. Задания открытого типа на дополнение (5 заданий)

5.1. Допишите команду для вывода строки «Привет, мир!»:

_____ ("Привет, мир!")

5.2. Допишите оператор для проверки условия «если $x > 0$ »:

_____ $x > 0$:

5.3. Допишите ключевое слово для определения функции:

_____ my_function():

5.4. Допишите метод списка для добавления элемента в конец:

my_list._____ (10)

5.5. Допишите ключевое слово для создания класса:

_____ Student:

6. Задания открытого типа с развернутым ответом (5 заданий)

6.1. Напишите программу на Python, которая запрашивает у пользователя два числа и выводит их сумму, разность, произведение и частное.

6.2. Напишите функцию factorial(n), вычисляющую факториал числа с помощью рекурсии. Приведите пример вызова.

6.3. Напишите программу, которая выводит все чётные числа от 1 до 20 с помощью цикла for.

6.4. Напишите программу, которая запрашивает строку и выводит её в обратном порядке, а также количество символов в ней.

6.5. Создайте класс Student с полями name и age, конструктором и методом info(), выводящим информацию о студенте. Приведите пример использования класса.

Ответы для проверки

1. Один правильный ответ

1.1 — Б

1.2 — В

1.3 — Б

1.4 — Б

1.5 — В

2. Несколько правильных ответов

2.1 — А, В, Д

2.2 — А, В, Д

2.3 — А, Б

2.4 — А, В, Д

2.5 — А, В, Д

3. Соответствие

3.1 — 1-Б, 2-А, 3-В, 4-Г

3.2 — 1-В, 2-А, 3-Б, 4-Г

3.3 — 1-Б, 2-А, 3-В, 4-Г

3.4 — 1-Б, 2-А, 3-В, 4-Г

3.5 — 1-Б, 2-А, 3-В, 4-Г

4. Последовательность

4.1 — 1, 2, 3, 4, 5

4.2 — 1, 2, 3, 4, 5

4.3 — 1, 2, 3, 4, 5

4.4 — 1, 2, 3, 4, 5

4.5 — 1, 2, 3, 4, 5

5. Дополнение

5.1 — print

5.2 — if

5.3 — def

5.4 — append

5.5 — class

6. Развернутый ответ (примеры)

6.1.

python

Копировать

Скачать

```
a = float(input("Введите первое число: "))
b = float(input("Введите второе число: "))
print("Сумма:", a + b)
print("Разность:", a - b)
print("Произведение:", a * b)
print("Частное:", a / b if b != 0 else "деление на ноль")
```

6.2.

python

Копировать

Скачать

```
def factorial(n):
    if n == 0 or n == 1:
        return 1
    return n * factorial(n - 1)
```

```
print(factorial(5)) # 120
```

6.3.

python

Копировать

Скачать

```
for i in range(1, 21):
    if i % 2 == 0:
        print(i)
```

6.4.

python

Копировать

Скачать

```
s = input("Введите строку: ")
print("Обратный порядок:", s[::-1])
print("Количество символов:", len(s))
```

6.5.

python

Копировать

Скачать

```
class Student:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def info(self):
        print(f"Студент: {self.name}, возраст: {self.age}")
```

```
s = Student("Иван", 20)
```

```
s.info()
```