

# Фонд оценочных средств (контрольно-измерительные материалы)

## Министерство просвещения Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«Красноярский государственный педагогический университет им. В.П. Астафьева»

### Институт математики, физики и информатики

(наименование института/факультета)

Кафедра-разработчик Информатики и информационных технологий в образовании  
(наименование кафедры)

УТВЕРЖДЕНО

на заседании кафедры

Протокол № 9

от «08» мая 2024 г.

ОДОБРЕНО

на заседании научно-методического совета

направления подготовки

Протокол № 7 от «15» мая 2024 г



Пак Н.И.



Аёшина Е.А.

## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля и промежуточной аттестации обучающихся

### «Программирование»

(наименование дисциплины/модуля/вида практики)

Направление подготовки:

*44.03.05 Педагогическое образование (с двумя профилями подготовки)*

направленность (профиль) образовательной программы

*Математика и информатика*

по очной форме обучения (общая трудоемкость дисциплины 7 з.е.)

Составители:

*канд. физ.-мат. наук, доцент кафедры ИИТвО Романов Д. В.*

### 3.3.1. Оценочное средство 1 «Вопросы и задания к зачёту»

- 1) Какой из циклов оптимальней использовать для повторения оператора(ов), если известно количество шагов цикла?
  - a) Цикл "while".
  - b) Цикл "for".
- 2) Чему равен результат вычисления выражения  $x + 3*b + x$  при  $x = 12$  и  $b = 8$ ?
  - a) 132
  - b) 48
  - c) 300
- 3) 1. Какой оператор используется для проверки равенства двух значений в Python?
  - a) =
  - b) ==
  - c) ===
  - d) !=
- 4) Какой тип данных в Python используется для хранения последовательности символов?
  - a) int
  - b) float
  - c) str
  - d) list
- 5) Какой из следующих операторов используется для добавления элемента в конец списка?
  - a) append()
  - b) insert()
  - c) extend()
  - d) add()
- 6) Какой результат вернёт выражение  $5 // 2$  в Python?
  - a) 2.5
  - b) 2
  - c) 3
  - d) 1
- 7) Какой тип данных в Python используется для хранения истинных или ложных значений?
  - a) int
  - b) float
  - c) str

- d) bool
- 8) **Какой из следующих методов используется для получения количества элементов в списке?**
- a) count()
  - b) size()
  - c) len()
  - d) length()
- 9) **Какой оператор используется для создания цикла с известным количеством повторений?**
- a) while
  - b) for
  - c) if
  - d) elif
- 10) **Какой из следующих методов можно использовать для удаления элемента из списка по его индексу?**
- a) remove()
  - b) pop()
  - c) delete()
  - d) discard()
- 11) **Какое ключевое слово используется для определения функции в Python?**
- a) func
  - b) define
  - c) def
  - d) lambda
- 12) **Какой результат вернёт выражение `3 ** 2` в Python?**
- a) 5
  - b) 6
  - c) 9
  - d) 8
- 13) **Что выведет следующий код: `print(type(15))`?**
- a) <class 'int'>
  - b) <class 'float'>
  - c) <class 'str'>
  - d) <class 'number'>
- 14) **Какой оператор используется для целочисленного деления?**
- a) /
  - b) //
  - c) %
  - d) \*\*

- 15) **Что делает оператор `**` в Python?**
- a) Умножение
  - b) Деление
  - c) Возведение в степень
  - d) Модуль
- 16) **Как создать список из пяти элементов, равных `None`?**
- a) `[None] * 5`
  - b) `[None for _ in range(5)]`
  - c) `None * 5`
  - d) `{None: 5}`
- 17) **Что выведет следующий код: `print("Python"[-1])`?**
- a) 'P'
  - b) 'n'
  - c) 't'
  - d) Ошибка
- 18) **Какой результат выполнения следующего кода: `len("123" + "456")`?**
- a) 3
  - b) 6
  - c) 123456
  - d) `TypeError`
- 19) **Что делает метод `append()` в списках Python?**
- a) Удаляет последний элемент списка
  - b) Добавляет элемент в начало списка
  - c) Добавляет элемент в конец списка
  - d) Сортирует список
- 20) **Какой тип данных используется для неизменяемой упорядоченной последовательности элементов?**
- a) `list`
  - b) `set`
  - c) `dict`
  - d) `tuple`
- 21) **Что вернет выражение `3 == 3.0`?**
- a) `True`
  - b) `False`
  - c) `TypeError`
  - d) `None`
- 22) **Что выведет следующий код?**
- ```
x = [1, 2, 3, 4]
print(x[1:3])
```

- a) [1, 2]
- b) [2, 3]
- c) [1, 2, 3]
- d) [2, 3, 4]

23) Какой результат будет у следующего выражения?

```
print(2 ** 3 ** 2)
```

- a) 64
- b) 512
- c) 256
- d) 8

24) Что выведет следующий код?

```
def func(x):  
    return x * x
```

```
print(func(3))
```

- a) 6
- b) 9
- c) 3
- d) Ошибка

25) Что делает следующий код?

```
for i in range(5):  
    if i == 2:  
        break  
    print(i)
```

- a) 0 1 2
- b) 0 1
- c) 0 1 2 3 4
- d) 1 2 3 4

26) Какой результат будет у следующего кода?

```
a = [1, 2, 3]  
b = a  
b.append(4)  
print(a)
```

- a) [1, 2, 3]
- b) [1, 2, 3, 4]
- c) [4, 2, 3]
- d) [1, 2, 3, 4, 4]

27) Что вернет функция filter() в следующем примере?

```
numbers = [1, 2, 3, 4, 5, 6]
```

```
result = list(filter(lambda x: x % 2 == 0, numbers))  
print(result)
```

- a) [1, 3, 5]
- b) [2, 4, 6]
- c) [1, 2, 3, 4, 5, 6]
- d) [2, 3, 4, 5, 6]

28) Какой результат будет у следующего кода?

```
def func(a, b=2, c=3):  
    return a + b + c
```

```
print(func(1, c=4))
```

- a) 7
- b) 6
- c) 5
- d) 10

29) Что выведет следующий код?

```
x = [1, 2, 3]  
y = x[:]  
y[0] = 0  
print(x)
```

- a) [0, 2, 3]
- b) [1, 2, 3]
- c) [0, 1, 2, 3]
- d) [0, 0, 3]

30) Какой результат будет у следующего кода?

```
x = [1, 2, 3]  
y = [4, 5, 6]  
z = x + y  
print(z)
```

- a) [1, 2, 3, 4, 5, 6]
- b) [5, 7, 9]
- c) [1, 2, 3, 10, 11, 12]
- d) Ошибка

31) Что выведет следующий код?

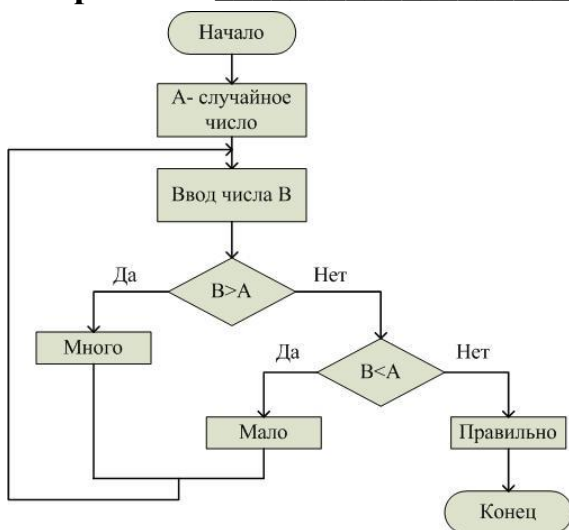
```
def func(x):  
    return [i * 2 for i in x]
```

```
print(func([1, 2, 3]))
```

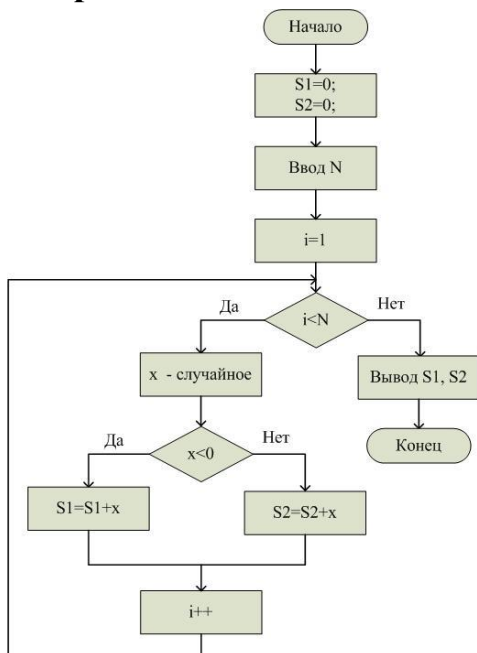
- a) [1, 2, 3]

- b) [2, 4, 6]
  - c) [2, 3, 4]
  - d) [0, 2, 4]
- 32) **Переменные, создаваемые в теле функции, называются локальными, потому что:**
- a) Они существуют только во время выполнения функции.
  - b) К ним можно обращаться только после их объявления.
  - c) Их имена должны начинаться с символа l.
- 33) **Отметьте допустимые имена функций:**
- a) `_This_Function_12_x_`
  - b) `ax%u7`
  - c) `ffffffAAAAAA`
  - d) `3_pi`
  - e) `calculateIt`
  - f) `thisname_is_too_long_forafunction`
- 34) **Как называется функция, которая вызывает саму себя?**
- a) конструктор
  - b) деструктор
  - c) подставляемая
  - d) рекурсивная
- 35) **В чем разница между фактическими и формальными параметрами?**
- a) Формальные параметры могут использоваться только вне тела функции, а фактические - используются как вне функции, так и внутри её.
  - b) Формальные параметры определены в теле функции, а фактические - значения, с которыми функция вызывается.
  - c) Нет различий.
- 36) **Какое из следующих утверждений об операторе return является верным?**
- a) Оператор return должен стоять последним в теле функции.
  - b) Оператор return завершает выполнение функции.
  - c) В теле функции должен присутствовать только один оператор return.
- 37) **Даны натуральные числа a, b, обозначающие соответственно числитель и знаменатель дроби. Сократить дробь, т.е. получить правильную несократимую дробь. Какую часть в предложенной задаче вы можете выделить как функцию? \_\_\_\_\_**
- 38) **Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу для деления дроби на дробь. Результат должен быть несократимой дробью. Какую часть в предложенной задаче вы можете выделить как подпрограмму? \_\_\_\_\_**

39) Проанализировав схему, предположите, решение какой задачи изображено. \_\_\_\_\_



40) Проанализировав схему, предположите, решение какой задачи изображено. \_\_\_\_\_





### 3.3.2. Вопросы к экзамену

- 1) Принципы структурного программирования. Основные алгоритмические конструкции.
- 2) Язык C++. Структура программы. Типы данных.
- 3) Язык C++. Стандартная библиотека. Ввод-вывод данных. Примеры.
- 4) Язык C++. Организация ветвлений (if, ?:). Оператор множественного выбора (switch). Пример.
- 5) Язык C++. Организация циклов с постусловием и предусловием. Примеры.
- 6) Язык C++. Цикл с параметром. Его особенности и возможности. Примеры.
- 7) Язык C++. Указатели. Операторы работы с указателями. Арифметика указателей. Ссылки. Примеры.
- 8) Язык C++. Пользовательские функции. Способы передачи данных в функцию. Примеры.
- 9) Язык C++. Статические одномерные массивы. Создание массивов и обработка элементов массивов.
- 10) Передача массивов в функцию. Примеры.
- 11) Язык C++. Динамические одномерные массивы. Создание одномерных динамических массивов и обработка элементов массивов. Передача массивов в функцию. Связь массивов и указателей. Примеры
- 12) Язык C++. Строки. Обработка строк. Связь строк, массивов и указателей. Пример.
- 13) Язык C++. Структуры. Описание структуры в программе. Массивы структур. Пример.
- 14) Язык C++. Динамические односвязные списки. Примеры процедур обработки списков.
- 15) Язык C++. Алгоритмы поиска и сортировки.
- 16) Основы анализа эффективности алгоритмов. Трудоёмкость алгоритмов. Временная сложность алгоритмов.
- 17) Язык Питон. Основные типы данных. Достоинства и недостатки. Примеры.
- 18) Язык Питон. Числовые типы данных. Операции. Свойства.
- 19) Язык Питон. Тип данных list, его возможности. Пример.
- 20) Язык Питон. Тип данных dict, его возможности. Пример.
- 21) Язык Питон. Тип данных str, его возможности. Пример.
- 22) Язык Питон. Тип данных set, его возможности. Пример.
- 23) Язык Питон и функциональное программирование. Декораторы. map. reduce. Примеры.

24) Сортировка в языке Питон. `sort`, `sorted`. Использование ключей сортировки.