

**федеральное государственное автономное образовательное учреждение высшего
образования
Первый Московский государственный медицинский университет им. И.М.
Сеченова Министерства здравоохранения Российской Федерации
(Сеченовский Университет)**

Институт цифрового биодизайна и искусственного интеллекта в медицине
Кафедра информационных технологий и обработки медицинских данных

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Интеллектуальные системы в социологии и медицине

основная профессиональная образовательная программа высшего
профессионального образования - программа бакалавриата

45.03.04 Интеллектуальные системы в гуманитарной сфере
Интеллектуальные системы в социологии и медицине

Перечень вопросов по курсу для проведения централизованного тестирования.
Правильные варианты ответов обозначены буквой «а».

1. Что выведет этот код? `print(type(5))`
 - a) класс `int`
 - b) класс `float`
 - c) класс `number`
 - d) класс `str`
2. Какой оператор используется для возведения в степень?
 - a) два символа звездочки
 - b) символ циркумфлекс
 - c) слово `pow`
 - d) два символа слэша
3. Как создать пустой список?
 - a) `list()`
 - b) квадратные скобки
 - c) `list[]`
 - d) `new list()`
4. Что такое `None` в Python?
 - a) Объект, обозначающий отсутствие значения
 - b) То же самое, что `False`
 - c) Пустая строка
 - d) Специальный тип ошибки
5. Какой из этих типов является изменяемым?
 - a) `list`
 - b) `int`
 - c) `str`
 - d) `tuple`
6. Что вернет выражение `3` умножить на строку `ab`?
 - a) `ababab`
 - b) `ab3`
 - c) `aaabbb`
 - d) Ошибку
7. Как получить остаток от деления `10` на `3`?
 - a) 10 процент 3
 - b) 10 два слэша 3
 - c) 10 слэш 3
 - d) `10 mod 3`
8. Что выведет `print(bool(строка False))`?
 - a) `True`
 - b) `False`
 - c) `None`

- d) Ошибку
9. Какой символ используется для однострочного комментария?
- a) решетка
 - b) два слэша
 - c) два дефиса
 - d) слэш звездочка
10. Что такое PEP 8?
- a) Руководство по стилю кода Python
 - b) Список новых функций Python
 - c) Документация по стандартной библиотеке
 - d) Условия лицензии Python
11. Какой метод используется для преобразования строки в нижний регистр?
- a) lower
 - b) casefold
 - c) tolower
 - d) small
12. Что выведет код: `print(строка hello с индексами от 1 до 4)`?
- a) ell
 - b) hel
 - c) ello
 - d) hell
13. Как объединить две строки a и b?
- a) a плюс b
 - b) a амперсанд b
 - c) a точка b
 - d) `concat(a, b)`
14. Что вернет строка `Python.find(строка th)`?
- a) 2
 - b) 3
 - c) 4
 - d) минус 1
15. Как разбить строку s на список по пробелам?
- a) `s.split()`
 - b) `s.split(пробел)`
 - c) `s.explode()`
 - d) `s.partition()`
16. Что выведет `print(len(строка с обратным слэшем и n))`?
- a) 2
 - b) 1
 - c) 0
 - d) 3

17. Какой метод проверяет, состоит ли строка только из цифр?
- a) `isdigit`
 - b) `isnumeric`
 - c) `isdecimal`
 - d) `isnumber`
18. Что делает метод `strip`?
- a) Удаляет пробелы и переносы строк с обоих концов
 - b) Удаляет все пробелы в строке
 - c) Разделяет строку на части
 - d) Удаляет только левые пробелы
19. Что вернет проверка `a` в строке `abc`?
- a) `True`
 - b) `False`
 - c) `1`
 - d) `0`
20. Как отформатировать строку: `Name: фигурные скобки, Age: фигурные скобки`?
- a) `format(Ivan, 25)`
 - b) `Ivan, 25`
 - c) `fill(Ivan, 25)`
 - d) `replace(Ivan, 25)`
21. Как добавить элемент `x` в конец списка `my_list`?
- a) `my_list.append(x)`
 - b) `my_list.add(x)`
 - c) `my_list.insert(x)`
 - d) `my_list.push(x)`
22. Что выведет `print([1, 2, 3] с шагом минус 1)`?
- a) `[3, 2, 1]`
 - b) `[1, 2, 3]`
 - c) `[]`
 - d) Ошибку
23. Как удалить последний элемент списка и вернуть его?
- a) `my_list.pop()`
 - b) `my_list.remove()`
 - c) `my_list.delete()`
 - d) `my_list.cut()`
24. Что вернет `len([[1, 2]] умножить на 2)`?
- a) `2`
 - b) `4`
 - c) `1`
 - d) `[2, 4]`
25. Как объединить два списка `a` и `b`?
- a) `a` плюс `b`

- b) `a.extend(b)`
- c) `a.append(b)`
- d) а вертикальная черта b

26. Что делает метод `list.insert(0, x)`?

- a) Добавляет x в начало списка
- b) Добавляет x в конец списка
- c) Заменяет первый элемент на x
- d) Ищет x по индексу 0

27. Как создать список чисел от 0 до 9?

- a) `list(range(10))`
- b) `range(10)`
- c) от 0 до 9 в квадратных скобках
- d) `list(0, 9)`

28. Что выведет `print([1, 2] is [1, 2])`?

- a) `False`
- b) `True`
- c) 1
- d) 0

29. Как отсортировать список `my_list` по убыванию?

- a) `my_list.sort(reverse равно True)`
- b) `my_list.sort(desc равно True)`
- c) `my_list.reversesort()`
- d) `my_list.descend()`

30. Чему равно `min([строка 10, строка 2, строка 30])`?

- a) 10
- b) 2
- c) 30
- d) 2 как число

31. Чем кортеж отличается от списка?

- a) Кортеж неизменяем
- b) Кортеж может хранить только числа
- c) Кортеж занимает больше памяти
- d) Кортеж не поддерживает индексацию

32. Как создать кортеж с одним элементом 5?

- a) (5, запятая)
- b) (5)
- c) `tuple(5)`
- d) [5]

33. Что вернет множество с элементами 1, 2, 2, 3?

- a) {1, 2, 3}
- b) {1, 2, 2, 3}
- c) {1, 3}

d) Ошибку

34. Какая операция для множеств возвращает элементы, которые есть в обоих множествах?

- a) intersection
- b) union
- c) difference
- d) symmetric_difference

35. Что выведет `print(len((1, 2, 3)))`?

- a) 3
- b) 1
- c) 6
- d) Ошибку

36. Как добавить элемент `x` в множество `s`?

- a) `s.add(x)`
- b) `s.append(x)`
- c) `s.insert(x)`
- d) `s.include(x)`

37. Что вернет `1` в `(1, 2, 3)`?

- a) `True`
- b) `False`
- c) `1`
- d) `0`

38. Как создать пустое множество?

- a) `set()`
- b) `{}`
- c) `[]`
- d) `()`

39. Что делает метод `set.discard(x)`?

- a) Удаляет `x` если он есть
- b) Удаляет `x` или вызывает ошибку
- c) Добавляет `x`
- d) Проверяет наличие `x`

40. Что вернет `(1, 2)` плюс `(3, 4)`?

- a) `(1, 2, 3, 4)`
- b) `(4, 6)`
- c) `(1, 2, (3, 4))`
- d) Ошибку

41. Как получить значение по ключу `key` из словаря `d`?

- a) `d[key]`
- b) `d.get(key)`
- c) `d.key`
- d) `d.find(key)`

42. Что делает метод `dict.get(key, default)`?
- a) Возвращает значение `key` или `default` если нет ключа
 - b) Возвращает значение `key` или вызывает ошибку
 - c) Добавляет ключ `key` со значением `default`
 - d) Удаляет ключ `key`
43. Как получить все ключи словаря `d`?
- a) `d.keys()`
 - b) `d.items()`
 - c) `d.values()`
 - d) `d.all_keys()`
44. Как добавить пару ключ-значение в словарь?
- a) `d[key]` равно `value`
 - b) `d.add(key, value)`
 - c) `d.insert(key, value)`
 - d) `d.append(key, value)`
45. Что вернет `len(пустой словарь)`?
- a) 0
 - b) 1
 - c) None
 - d) Ошибку
46. Как удалить ключ `key` из словаря `d` и вернуть его значение?
- a) `d.pop(key)`
 - b) `d.remove(key)`
 - c) `d.delete(key)`
 - d) `d.cut(key)`
47. Что выведет `print(список(словарь(a равно 1, b равно 2)))`?
- a) `[a, b]`
 - b) `[1, 2]`
 - c) `[(a, 1), (b, 2)]`
 - d) Ошибку
48. Могут ли ключи словаря быть списками?
- a) Нет
 - b) Да
 - c) Только если список пустой
 - d) Только в Python 2
49. Что делает метод `dict.update(другой словарь)`?
- a) Обновляет словарь добавляя пары из другого словаря
 - b) Заменяет словарь на другой
 - c) Сравнивает два словаря
 - d) Копирует словарь
50. Как создать словарь с ключами из списка и значениями 0?

- a) dict.fromkeys(список, 0)
- b) dict(список, 0)
- c) dict.create(список, 0)
- d) dict.init(список, 0)

51. Что выведет этот код? x равно 5; if x больше 3: print(Больше) else: print(Меньше)

- a) Больше
- b) Меньше
- c) Ошибку
- d) Ничего

52. Какой оператор проверяет равенство значений?

- a) два символа равно
- b) один символ равно
- c) три символа равно
- d) восклицательный знак и равно

53. Что делает оператор not?

- a) Возвращает противоположное булево значение
- b) Проверяет на None
- c) Отрицает число
- d) Удаляет переменную

54. Какой цикл выполняется пока условие истинно?

- a) while
- b) for
- c) loop
- d) do while

55. Что выведет: for i in range(3): print(i)

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

56. Как прервать выполнение цикла досрочно?

- a) break
- b) continue
- c) exit
- d) stop

57. Что делает оператор continue?

- a) Переходит к следующей итерации цикла
- b) Прерывает цикл
- c) Выходит из программы
- d) Продолжает выполнение после цикла

58. Что вернет 5 больше 3 и 2 меньше 1?

- a) False
- b) True

- c) None
- d) Ошибку

59. Как проверить, что x между 1 и 10 включительно?

- a) 1 меньше равно x меньше равно 10
- b) x в `range(1, 11)`
- c) x между 1 и 10
- d) 1 меньше x меньше 10

60. Что выведет: `i равно 0; while i меньше 3: i плюс равно 1; print(i)`

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

61. Как объявить функцию с именем `func`?

- a) `def func():`
- b) `function func():`
- c) `func def():`
- d) `define func():`

62. Что вернет функция без `return`?

- a) None
- b) 0
- c) Пустую строку
- d) Ошибку

63. Как сделать параметр функции необязательным?

- a) Указать значение по умолчанию
- b) Использовать ключевое слово `optional`
- c) Поместить его в конец
- d) Использовать символ вопроса

64. Что такое `args` в определении функции?

- a) Кортеж позиционных аргументов
- b) Словарь именованных аргументов
- c) Список аргументов
- d) Специальный параметр

65. Что такое `kwargs` в определении функции?

- a) Словарь именованных аргументов
- b) Кортеж позиционных аргументов
- c) Список аргументов
- d) Специальный параметр

66. Может ли функция возвращать несколько значений?

- a) Да, через кортеж
- b) Нет
- c) Только через список
- d) Только в Python 3

67. Что выведет: `def f(x, l равно []): l.append(x); return l; print(f(1)); print(f(2))`

- a) `[1] [1, 2]`
- b) `[1] [2]`
- c) `[1] [1]`
- d) Ошибку

68. Как передать аргументы по имени?

- a) `func(param равно value)`
- b) `func(param: value)`
- c) `func(param -> value)`
- d) `func(param value)`

69. Что делает оператор `return`?

- a) Возвращает значение и завершает функцию
- b) Выводит значение на экран
- c) Сохраняет значение в переменную
- d) Передает значение в другую функцию

70. Что такое локальная переменная?

- a) Переменная, определенная внутри функции
- b) Переменная, доступная везде
- c) Переменная с маленькой буквы
- d) Временная переменная

71. Как импортировать модуль `math`?

- a) `import math`
- b) `include math`
- c) `require math`
- d) `using math`

72. Как импортировать только функцию `sqrt` из `math`?

- a) `from math import sqrt`
- b) `import sqrt from math`
- c) `import math.sqrt`
- d) `from math include sqrt`

73. Что такое `name`?

- a) Имя текущего модуля
- b) Специальная функция
- c) Системная переменная
- d) Ключевое слово

74. Когда `name` равно `main`?

- a) Когда файл запущен напрямую
- b) Когда файл импортирован
- c) Всегда
- d) Никогда

75. Как создать псевдоним для импортированного модуля?

- a) `import math as m`
- b) `import math -> m`
- c) `import math: m`
- d) `import math from m`

76. Для чего используется конструкция `try-except`?

- a) Для обработки исключений
- b) Для проверки условий
- c) Для повторения операций
- d) Для импорта модулей

77. Как перехватить все исключения?

- a) `except Exception:`
- b) `except All:`
- c) `except *:`
- d) `except:`

78. Что делает блок `finally`?

- a) Выполняется всегда
- b) Выполняется при ошибке
- c) Выполняется при успехе
- d) Выполняется редко

79. Как вызвать исключение вручную?

- a) `raise`
- b) `throw`
- c) `exception`
- d) `error`

80. Что такое `ValueError`?

- a) Исключение при неверном значении
- b) Исключение при делении на ноль
- c) Исключение при импорте
- d) Исключение при памяти

81. Как открыть файл для чтения?

- a) `open(файл, r)`
- b) `open(файл, read)`
- c) `open(файл, w)`
- d) `open(файл, a)`

82. Какой режим открывает файл для записи?

- a) `w`
- b) `r`
- c) `a`
- d) `x`

83. Что делает метод `readline`?

- a) Читает одну строку
- b) Читает весь файл

- c) Читает указанное количество байт
- d) Читает все строки в список

84. Как гарантировать закрытие файла?

- a) Использовать with
- b) Вызвать close
- c) Использовать try-finally
- d) Любой из этих способов

85. Что делает метод write?

- a) Записывает строку в файл
- b) Читает из файла
- c) Перемещает указатель
- d) Закрывает файл

86. Что такое Python?

- a) Интерпретируемый язык программирования высокого уровня
- b) Компилируемый язык низкого уровня
- c) Язык разметки
- d) База данных

87. Какая основная философия Python?

- a) Должен быть только один очевидный способ сделать это
- b) Чем больше способов, тем лучше
- c) Главное скорость
- d) Главное минимализм

88. Что такое переменная в Python?

- a) Именованная область памяти для хранения данных
- b) Функция
- c) Модуль
- d) Класс

89. Какие основные типы данных в Python?

- a) Числа, строки, списки, словари
- b) Только числа
- c) Только строки
- d) Только списки

90. Что такое динамическая типизация?

- a) Тип переменной определяется во время выполнения
- b) Тип переменной задается явно
- c) Тип переменной нельзя изменить
- d) Тип переменной всегда строка

91. Что такое сильная типизация?

- a) Неявные преобразования типов запрещены
- b) Неявные преобразования разрешены
- c) Все типы одинаковы
- d) Нет проверки типов

92. Как узнать тип переменной?
- a) Функция `type`
 - b) Функция `typeof`
 - c) Функция `get_type`
 - d) Функция `check_type`
93. Какие числовые типы есть в Python?
- a) `int`, `float`, `complex`
 - b) Только `int`
 - c) Только `float`
 - d) `int`, `double`
94. Что такое целое число `int`?
- a) Число без дробной части
 - b) Число с дробной частью
 - c) Комплексное число
 - d) Дробное число
95. Что такое число с плавающей точкой `float`?
- a) Число с дробной частью
 - b) Целое число
 - c) Комплексное число
 - d) Строка
96. Что такое комплексное число?
- a) Число с действительной и мнимой частью
 - b) Простое число
 - c) Дробное число
 - d) Целое число
97. Как записать число 10 в двоичной системе?
- a) `0b1010`
 - b) `0x1010`
 - c) `0o1010`
 - d) `1010b`
98. Что такое строка в Python?
- a) Последовательность символов
 - b) Число
 - c) Список
 - d) Функция
99. Как создать многострочную строку?
- a) Использовать тройные кавычки
 - b) Использовать одинарные кавычки
 - c) Использовать двойные кавычки
 - d) Нельзя создать
100. Что такое индексация строк?

- a) Доступ к символу по номеру позиции
 - b) Изменение строки
 - c) Удаление строки
 - d) Создание строки
101. Что такое срез строк?
- a) Получение подстроки
 - b) Удаление строки
 - c) Изменение строки
 - d) Копирование строки
102. Что такое иммутабельность строк?
- a) Нельзя изменить после создания
 - b) Можно изменить после создания
 - c) Можно удалить
 - d) Можно копировать
103. Что такое список list?
- a) Упорядоченная изменяемая коллекция
 - b) Неупорядоченная коллекция
 - c) Неизменяемая коллекция
 - d) Упорядоченная неизменяемая коллекция
104. Как добавить элемент в конец списка?
- a) Метод append
 - b) Метод add
 - c) Метод insert
 - d) Метод push
105. Как удалить элемент из списка?
- a) Метод remove
 - b) Метод delete
 - c) Метод erase
 - d) Метод clear
106. Что такое список списков?
- a) Двумерный массив
 - b) Одномерный массив
 - c) Словарь
 - d) Множество
107. Как отсортировать список?
- a) Метод sort
 - b) Метод order
 - c) Метод arrange
 - d) Метод sorted
108. Что такое кортеж tuple?
- a) Упорядоченная неизменяемая коллекция
 - b) Упорядоченная изменяемая коллекция

- c) Неупорядоченная коллекция
 - d) Изменяемая коллекция
109. Чем кортеж отличается от списка?
- a) Кортеж неизменяем
 - b) Кортеж изменяем
 - c) Кортеж неупорядочен
 - d) Кортеж только для чисел
110. Когда использовать кортеж вместо списка?
- a) Для данных, которые не должны изменяться
 - b) Для данных, которые будут изменяться
 - c) Для неупорядоченных данных
 - d) Для больших данных
111. Как создать кортеж с одним элементом?
- a) Добавить запятую после элемента
 - b) Не добавлять запятую
 - c) Использовать список
 - d) Использовать словарь
112. Можно ли изменить элемент кортежа?
- a) Нет
 - b) Да
 - c) Только первый элемент
 - d) Только последний элемент
113. Что такое словарь dict?
- a) Неупорядоченная коллекция пар ключ-значение
 - b) Упорядоченная коллекция
 - c) Коллекция только чисел
 - d) Коллекция только строк
114. Что такое ключ в словаре?
- a) Уникальный идентификатор для значения
 - b) Значение
 - c) Метод
 - d) Функция
115. Какие типы могут быть ключами?
- a) Неизменяемые типы
 - b) Изменяемые типы
 - c) Любые типы
 - d) Только числа
116. Как получить значение по ключу?
- a) Через квадратные скобки
 - b) Через круглые скобки
 - c) Через фигурные скобки
 - d) Через угловые скобки

117. Что делает метод `get` у словаря?
- a) Возвращает значение или значение по умолчанию
 - b) Удаляет значение
 - c) Добавляет значение
 - d) Изменяет значение
118. Что такое множество `set`?
- a) Неупорядоченная коллекция уникальных элементов
 - b) Упорядоченная коллекция
 - c) Коллекция с дубликатами
 - d) Коллекция пар ключ-значение
119. Как создать пустое множество?
- a) Функция `set`
 - b) Фигурные скобки
 - c) Квадратные скобки
 - d) Круглые скобки
120. Что такое операции над множествами?
- a) Объединение, пересечение, разность
 - b) Сложение, вычитание
 - c) Умножение, деление
 - d) Индексация, срез
121. Как проверить наличие элемента во множестве?
- a) Оператор `in`
 - b) Оператор `has`
 - c) Оператор `contains`
 - d) Оператор `find`
122. Что такое `bool`?
- a) Логический тип `True` или `False`
 - b) Числовой тип
 - c) Строковый тип
 - d) Список
123. Какие значения считаются `False`?
- a) `False`, `None`, `0`, пустые коллекции
 - b) Только `False`
 - c) Только `0`
 - d) Только пустые строки
124. Что такое логические операторы?
- a) `and`, `or`, `not`
 - b) `+`, `-`, `*`
 - c) `>`, `<`, `==`
 - d) `in`, `is`
125. Как работает оператор `and`?

- a) Возвращает True если оба операнда True
 - b) Возвращает True если хотя бы один True
 - c) Возвращает противоположное значение
 - d) Сравнивает числа
126. Как работает оператор or?
- a) Возвращает True если хотя бы один операнд True
 - b) Возвращает True если оба True
 - c) Возвращает противоположное значение
 - d) Сравнивает строки
127. Что проверяет оператор is?
- a) Идентичность объектов
 - b) Равенство значений
 - c) Принадлежность
 - d) Неравенство
128. Что проверяет оператор ==?
- a) Равенство значений
 - b) Идентичность объектов
 - c) Принадлежность
 - d) Неравенство
129. Что проверяет оператор is?
- a) Идентичность объектов
 - b) Равенство значений
 - c) Принадлежность
 - d) Неравенство
130. В чем разница между == и is?
- a) == сравнивает значения, is сравнивает объекты
 - b) == сравнивает объекты, is сравнивает значения
 - c) Нет разницы
 - d) == для чисел, is для строк
131. Что такое if statement?
- a) Условное выполнение кода
 - b) Цикл
 - c) Функция
 - d) Класс
132. Что такое else?
- a) Выполняется если условие if ложно
 - b) Выполняется всегда
 - c) Выполняется если условие if истинно
 - d) Никогда не выполняется
133. Что такое elif?
- a) Проверка дополнительного условия
 - b) Конец условия

- c) Начало условия
 - d) Цикл
134. Можно ли писать вложенные if?
- a) Да
 - b) Нет
 - c) Только один уровень
 - d) Только два уровня
135. Что такое цикл for?
- a) Повторение для каждого элемента последовательности
 - b) Повторение пока условие истинно
 - c) Бесконечное повторение
 - d) Однократное выполнение
136. Что такое цикл while?
- a) Повторение пока условие истинно
 - b) Повторение для каждого элемента
 - c) Бесконечное повторение
 - d) Однократное выполнение
137. Что делает оператор break?
- a) Прерывает цикл
 - b) Переходит к следующей итерации
 - c) Начинает цикл заново
 - d) Пропускает цикл
138. Что делает оператор continue?
- a) Переходит к следующей итерации
 - b) Прерывает цикл
 - c) Начинает цикл заново
 - d) Пропускает цикл
139. Что такое else в цикле?
- a) Выполняется если цикл завершился нормально
 - b) Выполняется всегда
 - c) Выполняется если break не был вызван
 - d) Выполняется если был break
140. Что такое функция?
- a) Блок кода, который можно вызывать многократно
 - b) Переменная
 - c) Цикл
 - d) Условие
141. Как объявить функцию?
- a) Ключевое слово def
 - b) Ключевое слово function
 - c) Ключевое слово func
 - d) Ключевое слово define

142. Что такое параметры функции?
- a) Переменные в определении функции
 - b) Переменные при вызове функции
 - c) Результат функции
 - d) Имя функции
143. Что такое аргументы функции?
- a) Значения, передаваемые при вызове
 - b) Переменные в определении
 - c) Результат функции
 - d) Имя функции
144. Что такое возвращаемое значение?
- a) Результат работы функции
 - b) Параметры функции
 - c) Аргументы функции
 - d) Имя функции
145. Что такое локальная переменная?
- a) Переменная, определенная внутри функции
 - b) Переменная, определенная вне функций
 - c) Глобальная переменная
 - d) Встроенная переменная
146. Что такое глобальная переменная?
- a) Переменная, определенная на уровне модуля
 - b) Переменная внутри функции
 - c) Временная переменная
 - d) Аргумент функции
147. Как получить доступ к глобальной переменной внутри функции?
- a) Ключевое слово `global`
 - b) Ключевое слово `local`
 - c) Ключевое слово `nonlocal`
 - d) Автоматически
148. Что такое `nonlocal`?
- a) Для доступа к переменной из внешней функции
 - b) Для доступа к глобальной переменной
 - c) Для создания локальной переменной
 - d) Для удаления переменной
149. Что такое встроенная область видимости?
- a) Содержит встроенные функции и исключения
 - b) Содержит глобальные переменные
 - c) Содержит локальные переменные
 - d) Содержит временные переменные

150. Что такое число с плавающей точкой float?
- a) Число с дробной частью
 - b) Целое число
 - c) Комплексное число
 - d) Строка
151. Что такое комплексное число?
- a) Число с действительной и мнимой частью
 - b) Простое число
 - c) Дробное число
 - d) Целое число
152. Как записать число 10 в двоичной системе?
- a) 0b1010
 - b) 0x1010
 - c) 0o1010
 - d) 1010b
153. Что такое строка в Python?
- a) Последовательность символов
 - b) Число
 - c) Список
 - d) Функция
154. Как создать многострочную строку?
- a) Использовать тройные кавычки
 - b) Использовать одинарные кавычки
 - c) Использовать двойные кавычки
 - d) Нельзя создать
155. Что такое индексация строк?
- a) Доступ к символу по номеру позиции
 - b) Изменение строки
 - c) Удаление строки
 - d) Создание строки
156. Что такое срез строк?
- a) Получение подстроки
 - b) Удаление строки
 - c) Изменение строки
 - d) Копирование строки
157. Что такое иммутабельность строк?
- a) Нельзя изменить после создания
 - b) Можно изменить после создания
 - c) Можно удалить
 - d) Можно копировать
158. Что такое список list?
- a) Упорядоченная изменяемая коллекция

- b) Неупорядоченная коллекция
 - c) Неизменяемая коллекция
 - d) Упорядоченная неизменяемая коллекция
159. Как добавить элемент в конец списка?
- a) Метод append
 - b) Метод add
 - c) Метод insert
 - d) Метод push
160. Как удалить элемент из списка?
- a) Метод remove
 - b) Метод delete
 - c) Метод erase
 - d) Метод clear
161. Что такое список списков?
- a) Двумерный массив
 - b) Одномерный массив
 - c) Словарь
 - d) Множество
162. Как отсортировать список?
- a) Метод sort
 - b) Метод order
 - c) Метод arrange
 - d) Метод sorted
163. Что такое кортеж tuple?
- a) Упорядоченная неизменяемая коллекция
 - b) Упорядоченная изменяемая коллекция
 - c) Неупорядоченная коллекция
 - d) Изменяемая коллекция
164. Чем кортеж отличается от списка?
- a) Кортеж неизменяем
 - b) Кортеж изменяем
 - c) Кортеж неупорядочен
 - d) Кортеж только для чисел
165. Когда использовать кортеж вместо списка?
- a) Для данных, которые не должны изменяться
 - b) Для данных, которые будут изменяться
 - c) Для неупорядоченных данных
 - d) Для больших данных
166. Как создать кортеж с одним элементом?
- a) Добавить запятую после элемента
 - b) Не добавлять запятую
 - c) Использовать список

- d) Использовать словарь
167. Можно ли изменить элемент кортежа?
- a) Нет
 - b) Да
 - c) Только первый элемент
 - d) Только последний элемент
168. Чем кортеж отличается от списка?
- a) Кортеж неизменяем
 - b) Кортеж может хранить только числа
 - c) Кортеж занимает больше памяти
 - d) Кортеж не поддерживает индексацию
169. Как создать кортеж с одним элементом 5?
- a) (5, запятая)
 - b) (5)
 - c) tuple(5)
 - d) [5]
170. Что вернет множество с элементами 1, 2, 2, 3?
- a) {1, 2, 3}
 - b) {1, 2, 2, 3}
 - c) {1, 3}
 - d) Ошибку
171. Какая операция для множеств возвращает элементы, которые есть в обоих множествах?
- a) intersection
 - b) union
 - c) difference
 - d) symmetric_difference
172. Что выведет `print(len((1, 2, 3)))`?
- a) 3
 - b) 1
 - c) 6
 - d) Ошибку
173. Как добавить элемент `x` в множество `s`?
- a) `s.add(x)`
 - b) `s.append(x)`
 - c) `s.insert(x)`
 - d) `s.include(x)`
174. Что вернет `1` в `(1, 2, 3)`?
- a) True
 - b) False
 - c) 1
 - d) 0

175. Как создать пустое множество?
- a) `set()`
 - b) `{}`
 - c) `[]`
 - d) `()`
176. Что делает метод `set.discard(x)`?
- a) Удаляет `x` если он есть
 - b) Удаляет `x` или вызывает ошибку
 - c) Добавляет `x`
 - d) Проверяет наличие `x`
177. Что вернет `(1, 2)` плюс `(3, 4)`?
- a) `(1, 2, 3, 4)`
 - b) `(4, 6)`
 - c) `(1, 2, (3, 4))`
 - d) Ошибку
178. Как получить значение по ключу `key` из словаря `d`?
- a) `d[key]`
 - b) `d.get(key)`
 - c) `d.key`
 - d) `d.find(key)`
179. Что делает метод `dict.get(key, default)`?
- a) Возвращает значение `key` или `default` если нет ключа
 - b) Возвращает значение `key` или вызывает ошибку
 - c) Добавляет ключ `key` со значением `default`
 - d) Удаляет ключ `key`
180. Как получить все ключи словаря `d`?
- a) `d.keys()`
 - b) `d.items()`
 - c) `d.values()`
 - d) `d.all_keys()`
181. Как добавить пару ключ-значение в словарь?
- a) `d[key]` равно `value`
 - b) `d.add(key, value)`
 - c) `d.insert(key, value)`
 - d) `d.append(key, value)`
182. Что вернет `len(пустой словарь)`?
- a) `0`
 - b) `1`
 - c) `None`
 - d) Ошибку
183. Как удалить ключ `key` из словаря `d` и вернуть его значение?

- a) `d.pop(key)`
 - b) `d.remove(key)`
 - c) `d.delete(key)`
 - d) `d.cut(key)`
184. Что выведет `print(список(словарь(a равно 1, b равно 2)))`?
- a) `[a, b]`
 - b) `[1, 2]`
 - c) `[(a, 1), (b, 2)]`
 - d) Ошибку
185. Могут ли ключи словаря быть списками?
- a) Нет
 - b) Да
 - c) Только если список пустой
 - d) Только в Python 2
186. Что делает метод `dict.update(другой словарь)`?
- a) Обновляет словарь добавляя пары из другого словаря
 - b) Заменяет словарь на другой
 - c) Сравнивает два словаря
 - d) Копирует словарь
187. Как создать словарь с ключами из списка и значениями 0?
- a) `dict.fromkeys(список, 0)`
 - b) `dict(список, 0)`
 - c) `dict.create(список, 0)`
 - d) `dict.init(список, 0)`
188. Что выведет этот код? `x равно 5; if x больше 3: print(Больше) else: print(Меньше)`
- a) Больше
 - b) Меньше
 - c) Ошибку
 - d) Ничего
189. Какой оператор проверяет равенство значений?
- a) два символа равно
 - b) один символ равно
 - c) три символа равно
 - d) восклицательный знак и равно
190. Что делает оператор `not`?
- a) Возвращает противоположное булево значение
 - b) Проверяет на `None`
 - c) Отрицает число
 - d) Удаляет переменную
191. Какой цикл выполняется пока условие истинно?
- a) `while`

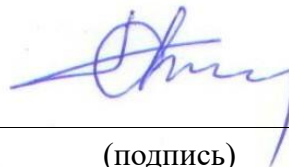
- b) for
 - c) loop
 - d) do while
192. Что выведет: `for i in range(3): print(i)`
- a) 0 1 2
 - b) 1 2 3
 - c) 0 1 2 3
 - d) 1 2
193. Как прервать выполнение цикла досрочно?
- a) break
 - b) continue
 - c) exit
 - d) stop

Принята на заседании кафедры Информационных технологий и обработки медицинских данных ЦЦМиИИМ ИЦБиИИМ

от «23» ноября 2024 г., протокол № 5

Заведующий кафедрой

Информационных технологий
и обработки медицинских
данных ЦЦМиИИМ



(подпись)

Лебедев Г.С.

(фамилия, инициалы)