

Список вопросов к экзамену и примеры экзаменационных билетов на экзамене 1 семестра (январь 2020 г.)

Перечень теоретических вопросов:

1. Определение информатики, понятия: информация, данные. Общая характеристика процесса сбора, передачи, обработки и накопления информации. (Лек.1, с.1-4)
2. Технические и программные средства реализации информационных процессов. (Лек.1, с. 5, 6)
3. Понятия информационной системы и информационной технологии. Классификация информационных систем. (Лек.1, с.7, 8)
4. Представление информации в машине. (Лек.1, с. 8-9)
5. Структурная схема ЭВМ. (Лек.1, с.10)
6. Основные этапы разработки программ. Цикл разработки программного обеспечения. (Лек.2, с.1,2 + расширение понятий в лек.2)
7. Алгоритм и его свойства. Способы отображения алгоритмов, их достоинства и недостатки. (Лек.2, с.11,12)
8. Спецификация данных в задаче. Класс, тип, структура данных. (Лек.2, с.5-7)
9. Базовые структуры в блок-схемах. Построение алгоритмов на основе базовых управляющих структур. (Лек.2, с.12,13 + лек.3 с. 14, 15)
10. Основные этапы спецификации задачи. (Лек.2, с.23, 24)
11. Основные виды тестирования. Методы структурного тестирования. (Лек.2, с. 15, 16)
12. Ключевые направления разработки функциональных тестов. Аномалии и альтернативные ситуации. (Лек.2, с.16-18 + лек.2 с.4)
13. Язык программирования Python. Интерпретаторы и компиляторы. Организация структурированной программы средствами языка Python. (Лек.3, с.2-5)
14. Простые типы данных. Базовые типы данных в Python. (Лек.3, с.7, 8)
15. Понятия: константа, число, переменная, объект в Python. Комментарии и строки документации в Python. (Лек.3, с.6, 8)
16. Операции над целыми и вещественными типами. Основные функции модуля *math* в Python. (Лек.3, с.10, 11 и 29)
17. Представление комплексных переменных и операции над ними. Основные функции модуля *cmath* в Python. (Лек.3, с.11 и 30)
18. Понятия: оператор, выражение, операнд. Основные операторы в Python (математические, сравнения, логические). (Лек.3, с.10-11, 13)
19. Приоритеты операторов в Python. Изменение приоритетов. Сокращенная запись операторов. (Лек.3, с.12, 13)
20. Итерационные и параметрические циклы. Два способа описания параметрического цикла в блок-схемах. (Лек.3, с.17, 19 + лек.4 с.1, 5)
21. Циклы с условием и с постусловием. Блок-схемы, реализующие эти циклы. (Лек.3, с.18, 19)
22. Операторы в Python: присваивания, условные операторы, оператор цикла *for..in..* и функция *range()*. (Лек.3, с.11 + лек.3 с.16, 17 + лек.4 с.2, 4, 5)
23. Операторы в Python: оператор цикла *while* и организация цикла с постусловием при помощи *while* и *break*. (Лек.5, с.11)
24. Структурированные типы. Массивы одномерные и двумерные. Статический массив. Реализация в Python (списки, кортежи). (Лек.4, с.5-7)
25. Структурированные типы. Операции над массивами средствами языка Python. Безопасное копирование массивов. (Лек.4, с.7-10)
26. Реализация матрицы в Python. Блок-схема вложенного параметрического цикла. Поэлементный ввод-вывод прямоугольной матрицы (пример). (Лек.4, с.12, 14 - 16)
27. Поиск экстремума и его индекса среди всех элементов массива. (Лек.4, с.17, 18)
28. Поиск экстремума среди элементов, удовлетворяющих условию. (Лек.4, с.18, 19)

29. Индексация для вырезанной области матрицы. (Лек.5, с.5, 6)
30. Связь между параметрическим и итерационным циклами на примере задачи поиска индекса последнего положительного элемента в массиве (блок-схемы). (Лек.5, с.12, 13)
31. Поиск в массиве (матрице) с досрочным выходом. Метод флажка. На примере поиска первого (или последнего) неотрицательного элемента. (Лек.5, с.16-18)
32. Цель, методология и принципы структурного программирования. Теорема Дейкстры о структурном программировании; понятие эквивалентности алгоритмов. (Лек.6, с.1-3)
33. Метод нисходящего проектирования, основные этапы. Проиллюстрировать решением задачи (**формулировку задачи спросить у лектора**) в форме блок-схемы головного модуля с указанием абстракций в ней. (Лек.6, с.5-7)
34. Вспомогательные алгоритмы: общего типа и функции. Реализация в Python и особенности оператора return. (Лек.6, с.12, 15-16)
35. Формальные параметры и фактические параметры (аргументы): правила записи и правила соответствия. (Лек.6, с.16, 17)
36. Классификация параметров в подпрограммах (входные, выходные, параметр-значение, параметр-переменная). (Лек.7, с.1-4)
37. Способы передачи аргументов: аргументы по умолчанию, переменное число аргументов. (Лек.7, с. 5-6)
38. Области видимости переменных. Локальные переменные. (Лек.7, с.7-9)
39. Резервированные слова *global* и *nonlocal*. (Лек.7, с. 9, 10)
40. Функции в качестве параметров подпрограмм. Проиллюстрировать примером программного кода. (Лек.7, с.11 - 13)
41. Многомодульные программы. Связь программных модулей. Области видимости. (Лек.8, с.2, 4, 5)
42. Правила размещения данных в общих областях. Переопределение данных и организация доступа к таким данным. Операторы *import*, *from ...import* и *from...import **. (Лек. 8, с. 3, 4, 6)
43. Типы файлов: текстовые и двоичные, примеры кодирования. Файлы прямого и последовательного доступа. (Лек. 8, с. 6 - 8, 10)
44. Работа с данными, хранящимися на внешнем устройстве. Ввод-вывод с помощью текстового файла. (Лек.8, с. 13-15)
45. Строки в Python. Операции со строками. Основные функции и функция *str()*. (Лек. 8, с. 10-12)
46. Форматный ввод/вывод. Спецификации формата: правила их записи и использования. (Лек. 8, с. 15-16)

Экзамен проводится в устной форме по билетам в виде письменной подготовки и изложения развернутого ответа.

Билет содержит два теоретических вопроса и одну задачу.

| | | |
|---|--|-------------------------------------|
| МЭИ | ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № | <i>Утверждаю: Зав. кафедрой</i> |
| | Кафедра <u>Прикладной математики</u> Дисциплина <u>Информатика</u> Институт <u>ИЭЭ</u> | «__» _____ 2020 г. |
| <ol style="list-style-type: none"> 1. Общая характеристика процесса сбора, передачи, обработки и накопления информации. Технические и программные средства реализации информационных процессов. 2. Правила размещения данных в общих областях. Переопределение (сокрытие) данных и организация доступа к таким данным. Операторы <i>import</i>, <i>from ...import</i> и <i>from...import *</i>. | | |

3. Разработать нисходящим способом (включая блок-схему головного модуля) и написать программный код с несколькими функциями для следующей задачи:

Заданы n точек плоскости своими координатами (x, y) .

Подсчитать процент точек, которые лежат на окружностях с центром в точке $(0, 0)$ и радиусами R_1 и R_2 . Для нахождения расстояния между точками использовать функцию. Предусмотреть альтернативную ситуацию.

| | | |
|------------|--|-------------------------------------|
| МЭИ | ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № | <i>Утверждаю: Зав. кафедрой</i> |
| | Кафедра <u>Прикладной математики</u> Дисциплина <u>Информатика</u> Институт <u>ИЭЭ</u> | |
| | | « » 2020 г. |

1. Структурная схема ЭВМ.
2. Области видимости переменных. Локальные переменные. Зарезервированные слова *global* и *nonlocal*.
3. Разработать нисходящим способом (включая блок-схему головного модуля) и написать программный код с несколькими функциями для следующей задачи:

Задана матрица $A(n \times m)$ из целых чисел. Если максимумы среди четных чисел в каждом из столбцов не превышают заданного параметра S , вывести сообщение: «Условие выполнено», иначе вывести сообщение: «Условие не выполнено». Для поиска максимума в столбце использовать функцию.

| | | |
|------------|--|-------------------------------------|
| МЭИ | ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № | <i>Утверждаю: Зав. кафедрой</i> |
| | Кафедра <u>Прикладной математики</u> Дисциплина <u>Информатика</u> Институт <u>ИЭЭ</u> | |
| | | « » 2020 г. |

1. Форматный ввод/вывод. Спецификации формата: правила их записи и использования.
2. Операции над целыми и вещественными типами. Основные функции модуля *math* в Python.
3. Разработать нисходящим способом (включая блок-схему головного модуля) и написать программный код для следующей задачи:

N цилиндров заданы параметрами: длина – L , диаметр – D .

Подсчитать суммарный объем цилиндров, для которых выполнено условие: объем цилиндра лежит в заданном диапазоне $[Va, Vb]$. Для нахождения объема цилиндра использовать функцию. Предусмотреть альтернативную ситуацию.

Требования к вопросу 3 (основной в билете)

1. Показать знание нисходящего проектирования:
 - 1) блок-схема только головного модуля с выделением подзадач с предусмотренными альтернативными ситуациями;
 - 2) код головного модуля с описанием заголовка функции (без описания тела функции) и вызовом функции;
 - 3) тело функции.
2. Не использовать глобальные переменные в функциях.
3. Не совмещать ввод-вывод исходных данных в одном цикле; вычисления и вывод результата внутри одной функции.
4. Можно ввод-вывод делать без использования файлов.
5. Матрицы и массивы должны выводиться поэлементно (можно без форматирования).
6. Показать знание структурного подхода:
 - 1) подключение модулей;
 - 2) переменные; здесь же выделяется память под массивы и матрицы, как исходные, так и если они формируются в результате работы функции (см. пример ниже).
 - 3) описания функций;
 - 4) операторы.

Массивы и матрицы по курсу – статические. Это означает, что **.append()** и динамическое выделение под произвольный размер – снижение оценки.

Пример. Пусть матрица **A** – исходное данное, а массив **B** будет получен в результате работы функции **f()**.

```
NMAX=10 # максимальная размерность обоих массивов

A=[0]*NMAX # целочисленная матрица максимальной размерности
for i in range(NMAX): A[i]=[0]*NMAX

B=[0]*NMAX # целочисленный массив максимальной размерности

na = 0 # реальный размер матрицы вводят с клавиатуры
nb = 0 # реальный размер массива

...операторы

nb = f(na, A, B) # вызов функции
```

Критерии выставления оценки на экзамене:

Оценка «ОТЛИЧНО» выставляется студенту, правильно выполнившему практическое задание, который показал при ответе на вопросы экзаменационного билета и на дополнительные вопросы, что владеет материалом изученной дисциплины, свободно применяет свои знания для объяснения различных явлений и решения задач.

Оценка «ХОРОШО» выставляется студенту, правильно выполнившему практическое задание и в основном правильно ответившему на вопросы экзаменационного билета и на дополнительные вопросы, но допустившему при этом не принципиальные ошибки.

Оценка «УДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который в ответах на вопросы экзаменационного билета допустил существенные и даже грубые ошибки, но затем исправил их сам, а также не выполнил практическое задание из экзаменационного билета, но либо наметил правильный путь его выполнения, либо по указанию экзаменатора решил другую задачу из того же раздела дисциплины.

Оценка «НЕУДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который:

- а) не ответил на вопросы экзаменационного билета и не смог решить, либо наметить правильный путь решения задачи из билета;
- б) не смог решить, либо наметить правильный путь решения задачи из экзаменационного билета и другой задачи на тот же раздел дисциплины, выданной взамен нее;
- в) при ответе на дополнительные вопросы обнаружил незнание большого раздела экзаменационной программы.