Лабораторная работа №5

По дисциплине: Основы программной инженерии Гробова Софья

ПИЖ-б-о-20-1

Программа indiv1.py, код и результат работы

Задача: С клавиатуры вводится число m (от 1 до 12). Вывести на экран название месяца, соответствующего цифре.

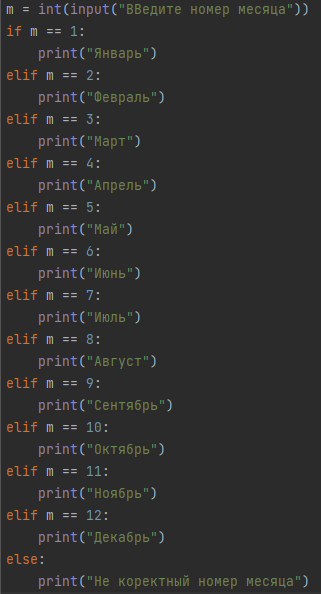


Рисунок 5.1 – Код программы indiv1.py

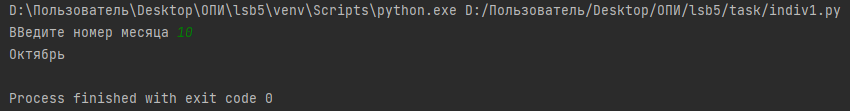


Рисунок 5.2 – Результат работы программы indiv1.py

Программа indiv2.py, код и результат работы

Задача: Провести исследование биквадратного уравнения, где a, b, c – действительные числа. Если действительных корней нет, то об этом должно быть выдано сообщение, иначе должны быть выданы 2 или 4 корня.

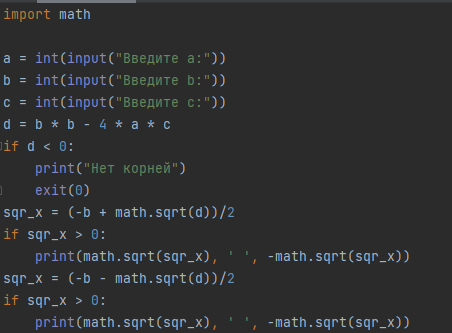


Рисунок 5.3 – Код программы indiv2.py

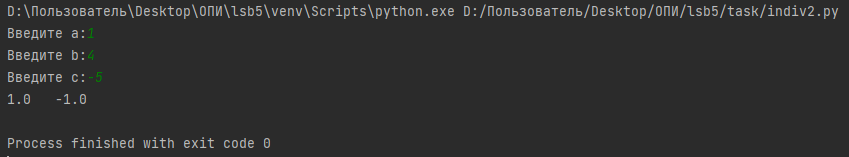


Рисунок 5.4 – Результат работы программы indiv2.py

Программа indiv3.py, код и результат работы

Задача: Определить среди всех двузначных чисел те, которые делятся на сумму своих цифр.

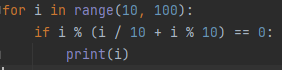


Рисунок 5.5 – Код программы indiv3.py

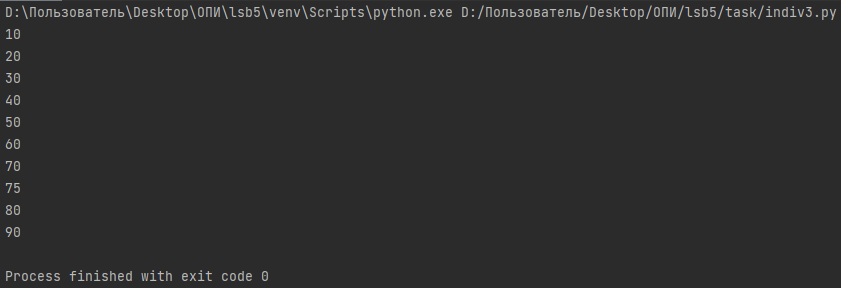


Рисунок 5.6 – Результат работы программы indiv3.py

Контрольные вопросы:

1. Для чего нужны диаграммы деятельности UML?

**Диаграммы деятельности** используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

1. Что такое состояние действия и состояние деятельности?

**Состояния деятельности** и **действия Состояние деятельности** (activity state)

- **состояние** в графе **деятельности**, которое служит для представления процедурной последовательности **действий**, требующих определенного времени.

1. Какие нотации существуют для обозначения переходов и ветвлений в диаграммах деятельности?

При построении диаграммы деятельности используются только нетриггерные переходы, т. е. такие, которые происходят сразу после завершения деятельности или выполнения соответствующего действия. Такой переход передает управление в последующее состояние сразу, как только закончится действие

или деятельность в предыдущем состоянии. На диаграмме такой переход изображается сплошной линией со

стрелкой.

Графически ветвление на диаграмме деятельности обозначается символом решения (decision), изображаемого в форме небольшого ромба, внутри которого нет никакого текста.

Для обозначений переходов используется стрелка указывающая на дальнейшее действие

1. Какой алгоритм является алгоритмом разветвляющейся структуры?

**Разветвляющийся алгоритм** – это **алгоритм**, в котором последовательность

выполнения операций зависит от определенных условий. Если в **алгоритме** присутствует

«действие1» и «действие2» (то есть ветвь 1 и ветвь 2), то это **разветвляющийся алгоритм** с полной альтернативой.

1. Чем отличается разветвляющийся алгоритм от линейного?

Линейный алгоритм - алгоритм, все этапы которого выполняются однократно и строго последовательно.

Разветвляющийся алгоритм - алгоритм, содержащий хотя бы одно условие, в результате проверки которого ЭВМ обеспечивает переход на один из двух возможных шагов.

1. Что такое условный оператор? Какие существуют его формы?

Условный оператор – оператор который проверяет значение одного значение с другим значением.

Существуют операторы if else elif

В операторах условий можно составлять сложные условия

1. Какие операторы сравнения используются в Python? Существуют операторы if else elif
2. Что называется простым условием? Приведите примеры. if a == b:
3. Что такое составное условие? Приведите примеры. if a == b and b not in list or a in list:
4. Какие логические операторы допускаются при составлении сложных условий? Коньюнкция, дизьюнкция, инверсия
5. Может ли оператор ветвления содержать внутри себя другие ветвления? Может
6. Какой алгоритм является алгоритмом циклической структуры?

**Алгоритм циклической структуры** – это **алгоритм**, в котором предусмотрено неоднократное выполнение одной и той же последовательности действий.

1. Типы циклов в языке Python. С предусловием

С щётчиком

Перебор элементов коллекций

1. Назовите назначение и способы применения функции range . Для определения промежутка чисел
2. Как с помощью функции range организовать перебор значений от 15 до 0 с шагом 2? for i in range(15, 0, 2):
3. Могул ли быть циклы вложенными? Могут
4. Как образуется бесконечный цикл и как выйти из него? While true

Выйти можно при помощи break

1. Для чего нужен оператор break ?

Для преждевременной остановки выполнения цикла

1. Где употребляется оператор continue и для чего он используется?

В циклах чтобы пропустить итерацию и приступить к следующей итерации

1. Для чего нужны стандартные потоки stdout и stderr?

stdin , **stdout и stderr** — три **потока** данных, созданные при запуске команды Linux.

1. Как в Python организовать вывод в стандартный поток stderr?

**import** sys

**class** MyStdout(object):

**def** init (self, old\_stdout): self.true\_sttdout = old\_stdout

**def** write(self, msg): self.true\_sttdout.write(msg[::-1])

**def** flush(self): self.true\_sttdout.flush()

sys.stdout = MyStdout(sys.stdout)

**print**("it works :)")

1. Каково назначение функции exit ?

Функция [exit()](https://docs-python.ru/standart-library/modul-sys-python/funktsija-exit-modulja-sys/) модуля [sys](https://docs-python.ru/standart-library/modul-sys-python/) - выход из Python.