

Программирование. Язык Python.

Введение.

Лабораторная работа № 1. Задачи.

Комплект 1: Установка среды программирования и разработки.

- 1.1: Установите Python последней версии <https://www.python.org/downloads/> и менеджер пакетов PIP <https://phoenixnap.com/kb/install-pip-windows> одним из предложенных в Интернете способов (или убедитесь, что PIP уже установлен).
- 1.2: Заведите аккаунт на <https://replit.com/> для лабораторных проектов/заданий на Python.

Комплект 2: Задачи для самостоятельного изучения материала.

- 2.1: Изучите документы по оформлению программного кода PEP8 <https://peps.python.org/pep-0008/>. Изучите документы по оформлению документации к коду <https://peps.python.org/pep-0257/>.
- 2.2: Изучите руководство Replit.com для начинающих: <https://techrocks.ru/2021/11/15/replit-beginners-guide/>.

Комплект 3: Задачи для самостоятельной работы.

- 3.1: Создайте простую программу калькулятор, которая позволяет из функции `main()` ввести два числа и тип арифметической операции, а потом вычисляет результат. Свой код опубликуйте на <https://replit.com/> и предоставьте ссылку в ответах на лабораторную работу в Moodle в документе-отчёте. Реализацию арифметических действий и вычисление результата с его возвратом сделайте в отдельной функции `calculate(...)`. Протестируйте свой калькулятор с помощью вызова нескольких своих простых функций `test_*`() с ключевым словом `assert` внутри. Обязательно напишите хорошую документацию к своему коду.

3.2: Реализуйте программно классическую простую игру "угадай число" (guess number) с помощью алгоритма медленного перебора (инкремента) по одному числу, либо с помощью алгоритма бинарного поиска. Алгоритм принимает на вход само число, которое он должен угадать, интервал значений в котором оно загадано и в цикле делает угадывания тем или иным выбранным вами способом. После угадывания из функции алгоритма возвращается угаданное число и число угадываний/сравнений, которые пришлось проделать. Обязательно напишите хорошую документацию к своему коду.