

**Вариант 10. В каждом задании обязательно использование модуля, которому посвящено задание.**

### **ЗАДАНИЕ 1. Модули PICKLE и DATETIME**

Выполняется на основе выполненного задания третьей темы.

- Необходимо создать текстовый файл и реализовать функцию логирования (без использования модуля logging). Функция должна вызываться из каждого метода ранее реализованных классов и записывать в файл строки следующего содержания: КЛЮЧ --- ДАТА И ВРЕМЯ --- КОММЕНТАРИЙ.  
Ключи: CRE (создание экземпляра класса), INF (изменение), ERR (сработало исключение).  
Комментарий: создано ..., удален ..., добавлен ..., распечатан ...
- Создать заполненные экземпляры реализованных класса, сериализовать их. В другом питоновском файле импортировать файл с описанием класса и десериализовать объекты. Применить к десериализованным объектам различные методы.

### **ЗАДАНИЕ 2. Модуль MATH**

- Реализовать две функции, вычисляющие математические формулы (файл math\_task\_X.png). Параметры формул являются аргументами функций.

### **ЗАДАНИЕ 3. Модуль RE**

Показать выполненное тестирование.

- Задача. URL

Вырезать доменную часть из URL

Ввод	Вывод
http://site.ru/folder/page.html	site.ru
http://si-te1.com/folder/page.html	si-te1.com
http://sub.site.ru/folder/page.html	sub.site.ru
https://sub.site.ru/folder/page.html	sub.site.ru
site.ru/folder/page.html	site.ru
site.ru	site.ru

### **ЗАДАНИЕ 4. Различные модули для работы с платформой и операционной системой**

- Собрать в папке файлы «task\_\*\*\*\*.py» – все ранее решенные задачи из тем А, В.
- Написать функцию, которая создаст папку «Ознакомительная папка» с двумя подпапками («тема А», «тема В»), переместит все файлы в правильные подпапки.
- Написать функцию, которая получает адрес ранее созданной папки «Ознакомительная папка» и выполнит обход всех подпапок и:
  - чтение всех «task\_\*\*\*\*.py» файлов, нахождение в тексте названия функции и параметров
  - программный запуск и выполнение данных файлов, подсчет времени выполнения

Результат работы функции:

```
folder "тема А":
>>> script "task_A111.py"
>>> >>> function "sum_cube(n, m)"
>>> >>> output "27 100 0 TEST PASSED"
>>> >>> time "1.135 sec"
>>> script "task_A222.py"
>>> >>> function "find_unique(lst)"
>>> >>> output "[2] [0.55, 0.66] TEST PASSED"
>>> >>> time "2.005 sec"
```