# Задания для самостоятельного выполнения

Выберите одну любую задачу из представленных ниже. Решите задачу, написав код на одном из следующих языков:

* Python
* C/C++
* C#
* Java

## Задача 1 (выполняется в ОС Linux)

Дан файл, содержащий текстовую строку – указание пути к дисковому устройству в системе Linux. Прочитайте файл и выведите на экран (stdout) следующую информацию о дисковом устройстве:

* Тип устройства, например:disk, part, lvm, rom;
* Общий объем в гигабайтах;
* В тех случаях, когда имеет смысл (например, если путь – это раздел диска), выведите также:
  + Объём свободного пространства в мегабайтах;
  + Тип файловой системы, например:ext4, swap;
  + Точку монтирования.

### Пример 1:

Входной файл:

/dev/sda

Вывод:

/dev/sda disk 64G

### Пример 2:

Входной файл:

/dev/sda1

Вывод:

/dev/sda1 part 1G 238M ext2 /boot

## Задача 2 (ОС Linux)

Напишите программу, выводящую информацию о системных сервисах и таймерах systemd.

Для системного сервиса выведите следующую информацию:

* Состояние сервиса (active, inactive);
* От какого пользователя и группы запускается сервис;
* Когда сервис был запущен в последний раз.

Для системного таймера выведите следующую информацию:

* Состояние таймера (active, inactive);
* Когда таймер был запущен в последний раз.

### Пример 1

Пример вызова: <your program> --timer apt-daily

Вывод:apt-daily.timer inactive, last started Mon 2019-07-01 11:23:24 MSK

### Пример2

Пример вызова: <your program> --service sshd

Вывод:sshd.serviceactive, user None, group None, last started Fri 2019-04-12 19:14:06 MSK

## Задача 3(любая ОС)

Дан файл, содержащий имена файлов, алгоритм хэширования (один из MD5/SHA1/SHA256) и соответствующие им хэш-суммы, вычисленные по соответствующему алгоритму и указанные в файле через пробел. Напишите программу, читающую данный файл и проверяющую целостность файлов.

### Пример

Файлсумм:

file\_01.bin md5 aaeab83fcc93cd3ab003fa8bfd8d8906

file\_02.binmd5 6dc2d05c8374293fe20bc4c22c236e2e

file\_03.bin md5 6dc2d05c8374293fe20bc4c22c236e2e

файл\_04.txt sha1 da39a3ee5e6b4b0d3255bfef95601890afd80709

Пример вызова:

<your program><path to the input file><path to the directory containing the files to check>

Формат вывода:

file\_01.bin OK

file\_02.bin FAIL

file\_03.bin NOT FOUND

файл\_04.txt OK

## Задача4(любая ОС)

Напишите прототип тестовой системы, состоящей из двух тест-кейсов.

Каждый тест-кейс имеет свой номер (tc\_id) и название (name); кроме того, каждый тест-кейс определяет отдельные методы (или функции, в зависимости от избранной модели реализации) для подготовки (prep), выполнения (run) и завершения (clean\_up) тестов. Метод execute задаёт общий порядок выполнения тест-кейса и обрабатывает исключительные ситуации.

### Тест-кейс 1: Список файлов

* **[prep]**Если текущее системное время, заданное как целое количество секунд от начала эпохи Unix, не кратно двум, то необходимо прервать выполнение тест-кейса.
* **[run]**Вывести список файлов из домашней директории текущего.
* **[clean\_up]**.

### Тест-кейс 2: Случайный файл

* **[prep]**Если объем оперативной памяти машины, на которой исполняется тест, меньше одного гигабайта, то необходимо прервать выполнение тест-кейса.
* **[run]**Создать файл test размером 1024 КБ с случайным содержимым.
* **[clean\_up]**Удалить файл test.

Все этапы выполнения тест-кейса, а также исключительные ситуации должны быть задокументированы в лог-файле или в стандартном выводе.