

# Задание 1. Программирование

## Описание задачи

Найти непрерывный подмассив в массиве, содержащий хотя бы одно число, который имеет наибольшую сумму.

## Условия

Необходимо написать программу с функцией `findMaxSubArray(A)`, принимающей на вход массив целых чисел  $A$  ненулевой длины и возвращающей непрерывный подмассив массива  $A$  ненулевой длины, который имеет наибольшую сумму среди всех непрерывных подмассивов массива  $A$ .

Язык программирования: python

Использование дополнительных библиотек и функций: не разрешается

В качестве решения необходимо прислать ссылку на github.

## Пример

На вход подается массив `[-2,1,-3,4,-1,2,1,-5,4]`

На выходе функции ожидается массив `[4,-1,2,1]`, имеющий максимальную сумму среди всех подмассивов равную 6.

## Задание 2. ML

### Описание задачи

Необходимо обучить нейросеть, способную по входному изображению лица определять пол человека на изображении.

### Данные

Для обучения и тестирования будет предоставлен набор из 100 тысяч картинок, из которых 50 тысяч будут содержать изображения лиц мужчин, а остальные 50 тысяч - изображения лиц женщин.

Данные можно скачать [отсюда](#). В загруженном архиве есть две папки, *male* и *female*, с изображениями лиц мужчин и женщин соответственно. Разбить данные на тренировочный и валидационный сет предлагается самостоятельно.

### Условия

В качестве результата необходимо прислать ссылку на github (можно объединить репозиторий с первой задачей), в котором должны быть:

1. Тренировочный скрипт или notebook-файл с кодом тренировки.  
Желательно, чтобы код был закомментирован;
2. Модель обученной нейросети (если модель большая - можно ссылкой на любой ресурс в описании);
3. Скрипт для использования нейросети, с помощью которого можно просчитать переданную через аргументы папку с изображениями. Скрипт должен сохранять файл *process\_results.json* с информацией о результатах процессинга.

Пример вызова:

```
python3 process.py folder/to/process/
```

Пример файла с результатами:

```
{ 'img_1.jpg': 'male', 'img_2.jpg': 'female', ... }
```

4. Описание решения, которое должно включать в себя описание процесса подготовки данных, используемой нейросети, параметров обучения, и полученных результатов. Так же в описании должна присутствовать инструкция для запуска тренировки и запуска нейросети.
5. В качестве фреймворка для обучения желательно использовать *pytorch*

### Оценка качества

Качество модели будет оцениваться по метрике accuracy.