Задача 3

1. Воспользуемся Python + Numpy для решения.

```python
def y(x):
    return 20 * x ** 2 - 2000 * x + 80000


xs = np.array([0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100])
ys = np.array([y(x) for x in xs])

print('Среднее x: ', xs.mean())
print('Среднее y: ', ys.mean())
```

[15] ✓ 0.0s

```
Среднее x:  50.0
Среднее y:  50000.0
```

2. Воспользуемся функцией модуля numpy corrcoef, которая подсчитает коэффициент корреляции между значениями двух массивов.

```python
np.corrcoef(xs, ys)
```

✓ 0.0s

```
array([[1., 0.],
       [0., 1.]])
```

3. Аналогичным способом получим коэффициент корреляции k ≈ -0.98 => величины обратнопропорционально изменяются с минимальной погрешностью.

```python
xs50 = np.array([0, 10, 20, 30, 40])
ys50 = np.array([y(x) for x in xs50])

np.corrcoef(xs50, ys50)
```

✓ 0.0s

```
array([[ 1.        , -0.98110491],
       [-0.98110491,  1.        ]])
```

Функция y на рассматриваемом промежутке убывает (парабола с ветвями вверх на значениях x < x-вершины), а функция x возрастает => две функции “обратнопропорциональны”, что подтверждает коэффициент корреляции

4. Переменная x - коэффициент, равны числу машин, работающих на бензине, на 1 человека

Переменная y - качество воздуха в городе.

При увеличении числа машин, работающих на бензине, увеличивается число выбросов вредных веществ в атмосферу => падает качество воздуха в городе

5. Новая функция y1(x) = 20 * y(x), то есть изменятся в постоянное число раз (y1(x) = c * y(x), c = 20). Следовательно, новая корреляция между величинами не изменится, так как новая функция имеет такой же вид зависимости от переменной x.

Проверим это, изменив функцию y(x) в питоне.

Получим, что новый коэффицент корреляции по-преженму равен 0

```python
def y(x):
    return x ** 2 - 100 * x + 4000


xs = np.array([0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100])
ys = np.array([y(x) for x in xs])

print('Среднее x: ', xs.mean())
print('Среднее y: ', ys.mean())
```

✓ 0.0s

```
Среднее x:  50.0
Среднее y:  2500.0
```

```python
np.corrcoef(xs, ys)
```

✓ 0.1s

```
array([[1., 0.],
       [0., 1.]])
```