

# Assessment: Coin Change Problem

## Problem 1

You have a set of coins of different denominations, and you need to deliver a specified amount of money as exact change. Write a function (in any program of your choice) that calculates the minimum number of coins required and which coins you will hand over.

If the change cannot be delivered with the available coins, you must return -1.

### Example 1

Input:

```
denominations = [0.25, 0.50, 1], amount = 2.25
```

Output:

```
minimum_coins: 3
```

```
coins_list: [1, 1, 0.25]
```

### Example 2

Input:

```
denominations = [0.50], amount = 0.75
```

Output:

```
minimum_coins: -1
```

```
coins_list: -1
```

Note: Infinite stock of each denomination can be assumed.

## Problem 2

Modify your previous program to now have a limited stock of each type of coin. For this, you can assume a new input that indicates the number of available coins of each type.

### Example 1

Input:

```
denominations = [0.25, 0.50, 1], available = [10, 5, 2], amount = 2.25
```

Output:

```
minimum_coins: 3
```

```
coins_list: [1, 1, 0.25]
```

### Example 2

Input:

```
denominations = [0.25, 0.50, 1], available = [20, 3, 1], amount = 2.75
```

Output:

```
minimum_coins: 4
```

```
coins_list: [1, 0.50, 0.50, 0.50, 0.25]
```