

## Algorithmen und Datenstrukturen

Exercise sheet 2

---

### 1 Aktienkurse

In dieser Aufgabe soll eine Methode implementiert werden, welche als Parameter einen Array enthält. Dieser Array beinhaltet die Aktienpreise eines bestimmten Unternehmen für einen Tag. Der Arrayindex ist dabei die Minute seit Start (Börse öffnet um 09:00 und schliesst um 17:30).

Beispiele:

```
stockprices[0] = 100 // Wert der Aktie um 09:00
stockprices[1] = 101 // Wert der Aktie um 09:01
stockprices[10] = 89 // Wert der Aktie um 09:10
stockprices[70] = 75 // Wert der Aktie um 10:10
stockprices[440] = 110 // Wert der Aktie um 16:20
```

Finden Sie den maximalen Gewinn der an diesem Tag erzielt werden konnte.

**ACHTUNG:** "shorting" (Verkauf bevor Kauf) ist nicht erlaubt. Ebenso ist es nicht erlaubt die Aktien zum gleichen Zeitpunkt zu kaufen und zu verkaufen. Es kann durchaus möglich sein, dass der Gewinn negativ ist.

```
1 class Trader {
2 public:
3     static int calculateMaxProfit(const int *values, const int ARRAY_SIZE);
4 };
```

### 2 Grösstes Produkt

In dieser Aufgabe soll eine Methode implementiert werden, welche als Parameter einen Array mit Integer Werten enthält. Aus diesem Array soll nun das grösste Produkt gefunden werden, welches mit 3 Elementen aus diesem Array erstellt werden kann.

```
1 class ArrayUtil {
2 public:
3     static int highestProduct(int *values, const int ARRAY_SIZE);
4 };
```

### 3 Anagramm Check

In dieser Aufgabe soll eine Methode implementiert werden, welche als Parameter zwei Strings enthält. Die Methode liefert den Wert `true`, falls diese beiden Strings Anagramme sind. Ansonsten `false`.

```
1 class StringUtil {
2 public:
3     static bool anagramCheck(string s1, string s2);
4 };
```