

## Arrays – Aufgaben:

1. Entwickeln Sie eine Funktion, die Ihnen ein Array von Zahlen zurückgibt, das die Zahlen der Form  $2^n$  enthält. (Bis zu einem Parameter **n**)

zb.:  $n = 6$

Array: [ 1, 2, 4, 8, 16, 32]

Testen Sie die Funktion mit einem entsprechendem Testprogramm !

2. Entwickeln Sie ein Programm, das vom Benutzer Zahlen in ein Array einliest, bis dieses voll ist. Der Benutzer kann die Eingabe abbrechen, bevor das Array voll ist, indem er 0 eingibt. Achten Sie auf Fehlerprüfungen !

- a. Lesen Sie vom Benutzer Ganzzahlen ein.
- b. Lesen Sie vom Benutzer Strings ein.
- c. Lesen Sie Bool Werte ein.
- d. Lesen Sie Dezimalzahlen ein.

3.

- a. Entwickeln Sie eine Funktion, die ein Array zufällig mit Integer Werten zwischen 1 und 100 befüllt. Geben Sie mit einem Parameter **n** die Länge des Arrays an, das befüllt werden soll. Das Array soll von der Funktion zurückgegeben werden.
- b. Entwickeln Sie eine weitere Funktion namens `search`, die in ihrem Array (das Sie der Funktion `search` als Parameter mitgeben) nach einer Zahl zwischen 1 und 100 sucht, die sie dieser Funktion ebenfalls als Parameter mitgeben. Die Funktion `search` soll Ihnen zurückgeben, wie oft die gesuchte Zahl in ihrem Array vorkommt.

Die Funktion soll die Anzahl des Vorkommens der Zahl in dem Array zurückgeben.

zb. `int x = search(new[]{1, 3, 3, 2, 5}, 3) // x = 2 (3 kommt 2 mal vor)`

Verwenden Sie beide Funktionen in einem entsprechenden Testprogramm !

4. Entwickeln Sie eine Funktion, die Ihnen ein Array zurückgibt, das Strings enthält, die aus einer zufälligen Reihenfolge aus Zeichen von a-z bestehen.

Geben Sie mit dem *Parameter* **slen** die Länge der Strings an, die das Array enthalten soll.

Geben Sie mit dem *Parameter* **alen** die Anzahl der Strings an, die das Array enthalten soll.

Die Funktion soll das Array *zurückgeben*.

zb.: `slen = 4, alen = 3`

Array: ["xejf", "kdtm", "etvh"]

Verwenden Sie den Befehl:

```
((char)<charcode>).ToString()
```

um einen String aus dem Charcode mit der Nummer <charcode> zu erzeugen.

zB.:

```
string s = ((char)97).ToString(); // s = "a"
```

5. Entwickeln Sie ein Programm, das eine Lottoziehung simuliert. Ihr Programm soll beim Starten ein 6-Stelliges Array mit zufälligen Zahlen zwischen 1 und 45 befüllen. Lesen Sie danach vom Benutzer 6 Zahlen ein. Vergleichen Sie diese Zahlen mit den Zahlen im Array und geben Sie aus, was der Benutzer erspielt hat . (3er, 4er, kein Gewinn ...) Geben Sie zusätzlich die Gewinnzahlen aus.