

# Programmieren 1 Aufgabenserie 7

Vithusan Ramalingam (21-105-515)

Jara Zihlmann (20-117-032)

Jan Ellenberger (21-103-643)

## **Aufgabe 1.)**

- a.) Unterschiedlich
- b.) Gleich
- c.) Gleich
- d.) Unterschiedlich
- e.) Unterschiedlich
- f.) Gleich

## **Aufgabe 2.)**

for i = 0 bis i = 100

modFive = i mod 5

modTree = i mod 3

if (modThreeResult == 0)

print "Bizz"

if (modFiveResult == 0)

print "Buzz"

if (modFiveResult == 0 und modThreeResult == 0)

print "BizzBuzz"

else

print i

### Aufgabe 3.)

function binaryAdder(Array A, Array B)

Array result length A + 1

for i = (length A - 1) ... 0

ab = A[i] + B[i] + result[i]

switch ab

case 0

result[i] = 0

case 1

result[i] = 1

case 2

result[i] = 0

result[i-1] = 1

case 3

result[i] = 1

result[i-1] = 1

Rückgabe result

#### Aufgabe 4.)

```
public static void shuffle(List<int> list) {  
    int n = list.size();  
    for (int i = n; i > 0; i--) {  
        Random random = new Random();  
        int r = random.nextInt(i+1);  
        this.swapElements(list, i, r);  
    }  
    return list;  
}  
  
public static swapElements(List<int> list, int i, int r) {  
    int memory = list.get(i);  
    list.set(i, list.get(r));  
    list.set(r, memory);  
}
```

#### Aufgabe 5.)

Ruby 1.1

Python 1.1

12345

#### Aufgabe 6.)

37

72

5

**Aufgabe 7.)**

72(Head)

37

15

**Aufgabe 8.)**

```
public boolean set(int index, Object object) {  
    if(index >= this.listElements.length - 1)  
        return false;  
    this.listElements[index] = object;  
    return true;  
}
```

**Aufgabe 9.)**

```
public int size() {  
    int size = 0;  
    Node<E> currentNode = this.startNode;  
    while (currentNode.getNext() != null) {  
        currentNode = currentNode.getNext();  
        size++;  
    }  
    return size;  
}
```

**Aufgabe 10.)**

Weil nicht bei jeder Implementierung von Collection dieselben Resultate von get und set erwartet werden. Daher ist es Sinnvoll dass jede Implementierung die get und set Methoden selber definiert sonst setzt oder erhält man falsche Werte. Als Beispiel benützt Stack die Rückgabe des letzten Elements, und List hingegen funktioniert über einen Index für get und set.