

Algorithmen und Datenstrukturen Übung 1

1 Aufgabe 1.1

1.1 (a)

```
int min = 1;
for (int i = 2; i < n; i++)
    if A[i] < A[min]
        min = i;
return min;
```

1.2 (b)

Der Algorithmus muss im schlimmsten Fall $4n+1$ Rechenschritte durchführen.

1.3 (c)

Der Algorithmus muss immer die gleiche Anzahl an Rechenschritte durchführen, da selbst wenn die Zahl in Index 1 die Kleinste ist müssen alle andere Zahlen überprüft werden.

2 Aufgabe 1.2

```
for (int i = 1; i < n; i++){
    min = FINDMINIMUM(A[i..n]);
    if (i != min){
        int temp = A[i];
        A[i] = A[min];
        A[min] = temp;
    }
}
```

re Der Algorithmus benötigt ungefähr $n(5 + n) = 5n + n^2$ Rechenschritte um im schlimmsten Fall das Array zu sortieren

3 Aufgabe 1.3

Übergeben werden die größte und kleinste Indizes

```
BINSEARCH(int lower, int upper, int x){
    int mid = roundDown(upper+lower);
    if (upper < lower){
        return 0;
    } else if (A[mid] == x){
        while (A[mid-1] == x){
            mid-1;
        }
        return mid;
    }
```

```
    } else if (A[mid] > x){  
        BINSEARCH(lower, mid-1, x);  
    } else {  
        BINSEARCH(mid+1, upper, x);  
    }  
}
```