

Tema NR 2: Algoritmanalys

Annika Svedin **ansv9785**
Felix Törnqvist **fetr0498**

24 januari 2017

Muntafrågor

1. Hur effektiv är den här sorteringsalgoritmen? Beräkna dess ordo-värde och visa hur du kommer fram till det.

```
1  public int [] sort(int [] numbers) {  
2  
3      for(int i = 0; i < numbers.length; i++) {  
4          for(int j = i; j < numbers.length; j++) {  
5  
6              if (numbers[j] < numbers[i]) {  
7                  int tmp = numbers[i];  
8                  numbers[i] = numbers[j];  
9                  numbers[j] = tmp;  
10             }  
11  
12         }  
13     }
```

```

13 ||     }
14 ||     return numbers;

```

2. Hur effektiv är följande algoritm?

```

1 || public boolean equalTo(String a[], String b[]) {
2 ||     if (a.length != b.length) {
3 ||         return false;
4 ||     }
5 ||
6 ||     for (int i = 0; i < a.length; i++) {
7 ||         if (!(a[i].equals(b[i]))) {
8 ||             return false;
9 ||         }
10 ||    }
11 ||    return true;
12 || }

```

3. Följande kod ska föreställa ett dåligt exempel på en stack-implementation, som inte har $O(1)$ utan $O(N)$

```

1 ||     int [] stack = new int [100];
2 ||     int cursor;
3 ||
4 ||     public void push(int element) {
5 ||         if (stack.length == cursor) {
6 ||             int [] newStack = new int [(stack.length * 2)];
7 ||
8 ||             for (int i = 0; i < stack.length; i++) {
9 ||                 newStack[i] = stack[i];

```

```

10 |         }
11 |         stack = newStack;
12 |     }
13 |     stack[cursor] = element;
14 |     cursor++;
15 | }

```

4. pusha stack, volym 2

```

1 |     private Node last;
2 |
3 |     public void push(int element) {
4 |         Node newNode = new Node(element);
5 |         last.next = newNode;
6 |         last = newNode;
7 |     }
8 |
9 |     private class Node {
10 |
11 |         private int element;
12 |         public Node next;
13 |
14 |         public Node(int element) {
15 |             this.element = element;
16 |         }
17 |     }

```