(b) (4 points) We are solving a problem about graph, and the problem can be solved by an algorithm, that looks like the following. (Q is a stack, $s \in V$ is some constant node.)

```
F(V, E)
  Q:=V
  ...
  while Q ≠ ∅ do
    v:=Q.Pop()
    for u ∈ v.Adj
      if (u, s) ∈ E
        ...
      endif
    endfor
  endwhile
```

Assume that everything that is omitted (marked by $\cdots$) is constant time, and the assignment $Q{:=}V$ is $\Theta(V)$. Analyse the time consumption for this algorithm for both matrices and lists.

5. In the following assignment, T_1 and T_2 are priority queues and A is an array.

```
Bum(A[1, ..., n])
  T1:=∅
  T2:=∅
  T2.insert(A[1])
  for k:=2 to n do
    if A[k] < A[k − 1] then
      T1.insert(A[k])
      A[k]:=T2.max()
    else
      T2.insert(A[k])
    endif
  endfor
```

(a) (4 points) Analyse the time consumption of the algorithm.

(b) (4 points) Argue, using an invariant, that after the program is run, the resulting array A is sorted.

teht.	1	2	3	4	5	yht.
max.	10	8	8	8	8	42
op.						

OHJ-2156 Analysis of Algorithms 29.10.2013 Henri Hansen

No calculators or books. Submit your answers on a separate paper.

1. Consider the following algorithm. Assume $A[1...n]$ is an array that contains integers.

```
Mort(A, n)
  for i = 2 to n do
    j:=1
    while j < i ∧ A[i] ≥ A[j] do
      j:=j + 1
    endwhile
    x:=A[j]; A[j]:=A[i]
    for k = j + 1 to i do
      y:=A[k]; A[k]:=x; x:=y
    endfor
  endfor
```

   (a) (2 points) Write an invariant for the *while-loop* inside.
   (b) (2 points) Write an invariant for the *for-loop*.
   (c) (2 points) Deduce what the algorithm does, justify using the invariants
   (d) (2 points) What is the worst case time consumption of the algorithm, and in what situation does this happen?
   (e) (2 points) What about the best case time consumption?

2. Give the Θ, or O and Ω- class of the following functions. Be as accurate as possible.
   (a) (2 points) $\sum_{i=1}^{n} i$
   (b) (2 points) $\log(\sum_{i=1}^{n} i)$
   (c) (2 points) $O(n \log n) + \Theta(n^2)$
   (d) (2 points) $n^2 \log n + n^3$

3. Solve the following recurrences, assumin $T(n)$ is a constant for $n < 2$. Simply give the answer in Θ.
   (a) (2 points) $T(n) = T(n-1) + \sqrt{n}$
   (b) (2 points) $T(n) = T(n/2) + n$
   (c) (2 points) $T(n) = 3T(n/3) + n$
   (d) (2 points) $T(n) = 4T(n/2) + n$

4. Let (V, E) be a directed graph. We wish to represent the graph by either using adjacency matrix or adjacency list. A list requires 32 bits for each vertex and 64 bits for each edge. A matrix, on the other hand, requires only 8 bits for each slot.
   (a) (4 points) Describe the situation where memory requirements for the two representations are equal.