# Algorithm assignment II

## Question 1: Heap Sort

assume the heap is Max heap

### When a[1]<a[2]<a[3]<....<a[n]

Build the heap:from n/2 to n assume n=$2^k$,k=$log_2(n)$

Then the Tree has k+1 layers. We will have to exchange the elements for$1+2+3+\ldots+k$ times to build the initial Max-heap.

Generally, we will need to exchange for$\Theta(log_2(n)^2)$ times to build the initial Heap.

Then we have to maintain the Max-heap.

final result is $\Theta$(n$\times log_2(n)$)

### When a[1]>a[2]>a[3]>...>a[n]

It is already a MAX-heap,so it takes no time to build the MAX-heap

Then it will take O(n$log_2(n)$)to maintain the Max_heap

## Question2:quick sort

### (a)

assume the H is the max-height of the Tree and h is the minimize height of the Tree.

We have following relationships:

$1 \div \alpha^h = n$

$a \div (1-\alpha)^H = n$

The result is $h = -lgn \div lg\alpha$

$H = -lgn \div lg(1-\alpha)$

### (b)

假设产生两个不同的1划分$\alpha, 1-\alpha$和$\theta, 1-\theta$的概率相同

那么比$\alpha, 1-\alpha$更好的划分必须更加接近中心1/2

概率为$(1 \div 2 - \alpha) \div (1 \div 2) = (1 - 2 \times \alpha)$