David Wang
CSE 100

Homework 2

1.) Heap-Delete(A, i):

```
if (A[i] < A[A.heap-size]
    Heap-Increase-Key(A, i, A[A.heap-size])
    A.heap-size--
else
    A[i] = A[heap-size]
    A.heap-size--
    Max-Heapify(A, i)
```

2.) QuickSort

part 1: When all elements have the same value, r is returned.

Modifying partition,

~~Partition(A, p, r)~~
~~x = A[r]~~
~~i = p - 1~~
~~count = 0~~
~~for~~

```
Partition(A, p, r)
{ x = A[r]
  i = p - 1
  count = 0
  for j = p to j = r
  { if A[j] == x
      count++
    if A[j] <= x
      i++
      swap(A[i], A[j])
  }
  swap(A[i+1], A[r])
  return (i+1-count)/2
}
```

Scanned with CamScanner

Exercise 2:

part 2:

The run time of Quicksort when all elements of A have the same value is $\Theta(n^2)$. The Partition would return index q=r, changing a problem with size n to size $n-1$.

Therefore $T(n) = T(n-1) + n$

Using the iteration method, $T(n) = \Theta(n^2)$

part 3:

The run time of Randomized-Quicksort would be $O(n)$ when all elements of array A have the same value.

For example if partioned into 2 equal subproblems,

$T(n) - 2N(n/2) = \Theta(n)$

Scanned with CamScanner

part 4:

```
Partition(A, P, r){
 int x = A[r]
swap(A[r], A[p])
int i = p-1
int k=p
for(int j=p+1 to r-1)
{     if(A[j]<x)
          i++;
          k=i+2;
          swap(A[i], A[j])
          swap(A[k], A[j])
   if(A[j]=x)
     k++;
    swap(A[k], A[j])
}
}
 swap(A[i+1], A[r])
 return (k+1) (i+1)
```

Modified so that elements will be moved after k

Scanned with CamScanner

5.) # loop invariant

1) the X is equal to A[-]

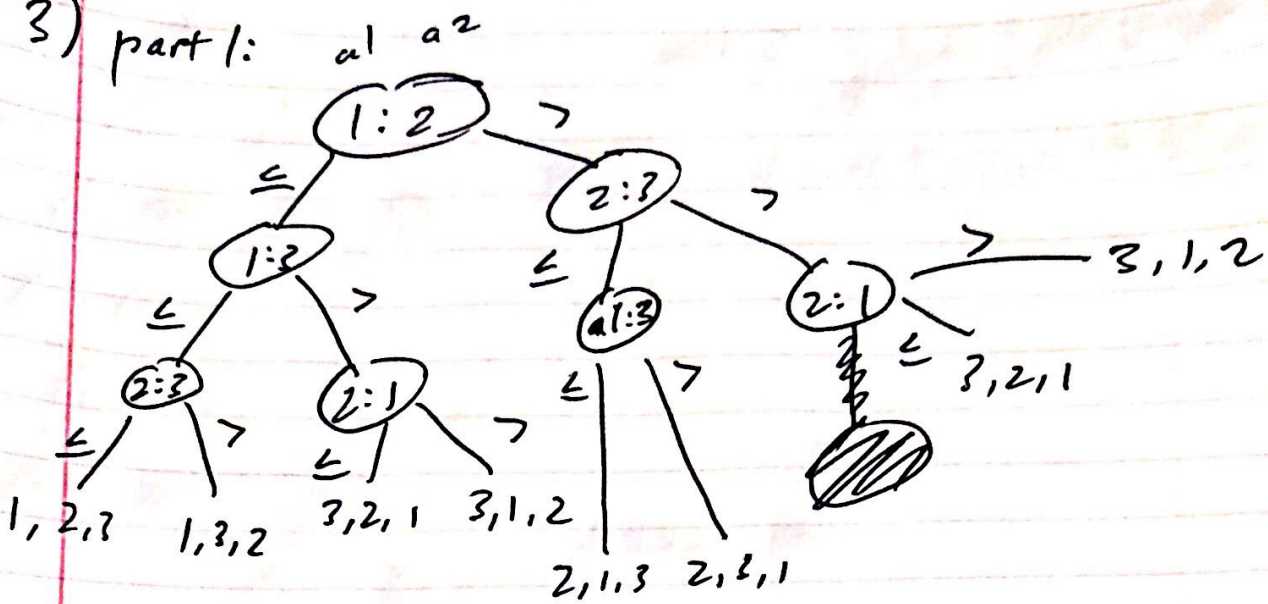
2) i is equal to p - 1

3.) k is equal to p

Scanned with CamScanner

$a1 = 1$
$a2 = 2$
$a3 = 3$

3) part 1:



part 3: The best case is finding the value on the first attempt.
Worst case is going through all nodes and finding it on the last try.

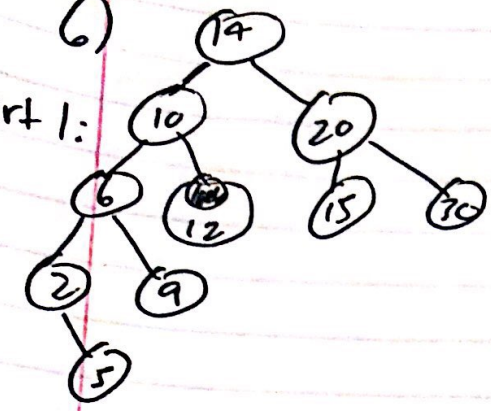
part 4:

Best case depth = 1
Worst case depth = 8

Scanned with CamScanner

6) 4h

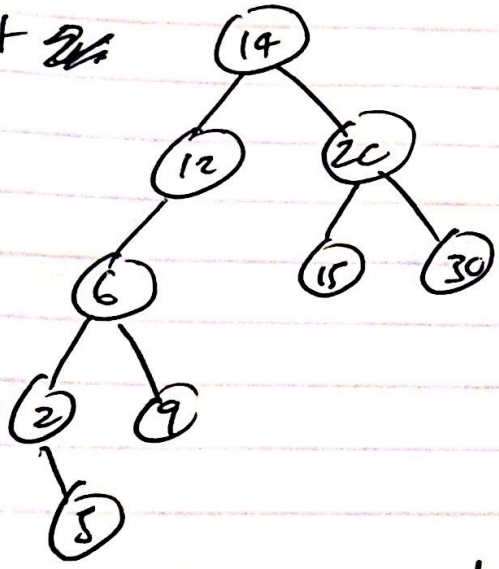
Part 1:



part 2: 2, ~~5, 6, 9, 10, 12, 14~~, 15, ~~20, 30~~

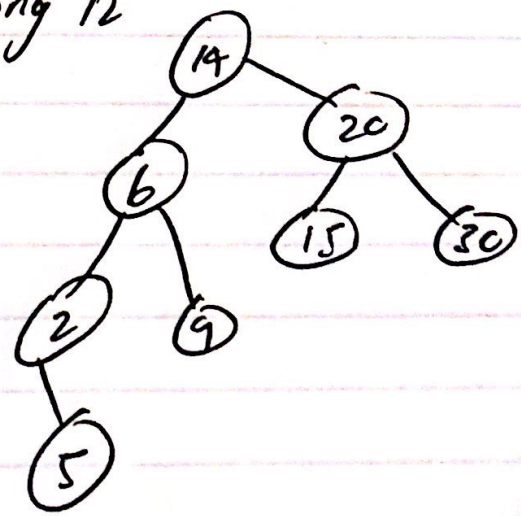
pg 2, 5, 6, 9, 10, 12, 14, 15, 20, 30

part 3:



deleting 10

deleting 12



Scanned with CamScanner