Homework 4, CPSC 4100-01, Winter 2017

1) Using Figure 7.1 as a model, illustrate the operation of PARTITION on the array $A[13,19,9,5,12,8,7,4,21,2,6,11]$. [CLRS 7.1-1]

5 points

2) Assume that we have an array with length n containing integer numbers. Develop an efficient algorithm that put all the negative numbers on the right side of the array and all the positive numbers on the left side. As an example:

A = [5 , -2 , 3 , 4 , -1 , 0 , -2 , 2]

Answer = [5 , 3 , 4 , 0 , 2 , -2 , -1 , -2]

Order of the items in the answer are not important.

15 points

3) We can build a heap by repeatedly calling MAX-HEAP-INSERT to insert the ele- ments into the heap. Consider the following variation on the BUILD-MAX-HEAP procedure:

```
BUILD-MAX-HEAP'(A)
A.heap-size = 1
for i = 2 to A.length
    MAX-HEAP-INSERT(A, A[i])
```

a) Do the procedures $BUILD\_MAX\_HEAP$ and $BUILD\_MAX\_HEAP'$ always create the same heap when run on the same input array? Prove that they do, or provide a counterexample.

b) Show that in the worst case, $BUILD\_MAX\_HEAP'$ requires $\Theta(n \log n)$ time to build an n-element heap.

```
MAX-HEAP-INSERT(A, key)
A.heap-size = A.heap-size + 1
A[A.heap-size] = -∞
HEAP-INCREASE-KEY(A, A.heap-size, key)
```

```
HEAP-INCREASE-KEY(A, i, key)
if key < A[i]
    error "new key is smaller than current key"
A[i] = key
while i > 1 and A[PARENT(i)] < A[i]
    exchange A[i] with A[PARENT(i)]
    i = PARENT(i)
```

[CLRS 6-1]

15 points

4) Implement the Strassen's method for matrix multiplication, to multiply two $n \times n$ matrices. Test your code for three example: a 2×2 case, a 4×4 case, and an 8×8 case. Report your code as well as your test cases.

15 points