UNIVERSITY OF GHANA

BACHELOR OF SCIENCE IN ENGINEERING

SECOND SEMESTER EXAMINATIONS: 2015/2016

DEPARTMENT OF COMPUTER ENGINEERING

CPEN 204: DATA STRUCTURES AND ALGORITHMS (3 Credits)

INSTRUCTION:
ANSWER ALL QUESTIONS FROM SECTIONS A AND B.

TIME ALLOWED: THREE (3) HOURS

SECTION A
ANSWER ALL QUESTIONS IN THIS SECTION.

1. Indicate the results of the following operations when performed on a dynamic set S. Assume that a key value *k* is an element of S and a pointer *x* points to an element in S.
   a) *SEARCH (S, k)* ***[2 marks]***
   b) *INSERT (S, x)* ***[2 marks]***
   c) *DELETE (S,x)* ***[2 marks]***
   d) *MINIMUM (S)* ***[2 marks]***
   e) *SUCCESSOR (S,x)* ***[2 marks]***
   f) *PREDECESSOR (S,x)* ***[2 marks]***
2. Given the following structure definitions and declarations in Figure 1, find error(s), if any, in statements **a** to **d**, and correct them.

```
struct Xxx
{
    int aa;  double *bb; char *cc[30]
};
struct Yyy
{
    int *dd;  double ee;
}; struct Xxx mm, nn; struct Yyy pp, qq;
```

Figure 1

   a. mm=pp; ***[2 marks]***
   b. strcpy(nn.cc, "Trial"); ***[2 marks]***

A

c. Xxx.aa = 13; *[2 marks]*

d. Yyy.ee = 76.9; *[2 marks]*

SECTION B
ANSWER ALL QUESTIONS IN THIS SECTION.

3. Figure 1 shows a queue of at most 12 elements implemented using array *Q [1...13]* with *head* [Q] =7.

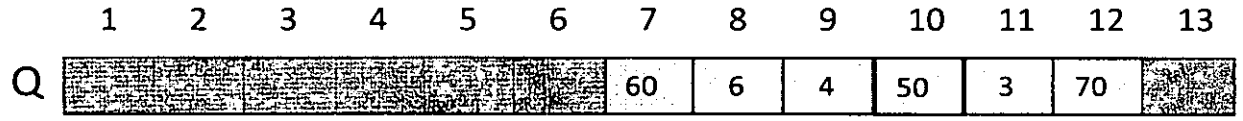

Figure 1

a. Illustrate the configuration of the queue after each of the following operations has been performed successively on it: ENQUEUE (Q, 3), ENQUEUE (Q, 1), ENQUEUE (Q, 4), DEQUEUE (Q), ENQUEUE (Q, 8), and DEQUEUE (Q).

[6 marks]

b. Find the value of *tail* [Q] after each of the operations in 3.(a) has been performed.

[3 marks]

c. Using diagrams, show how a **doubly linked list *L*** can be used to represent the dynamic set {1, 4, 9, 16}. **[5 marks]**

d. Show the configuration of ***L*** after performing the following operation:
LIST-INSERT (*L, x*), where *key[x] =25.* **[2 marks]**

e. Using diagrams describe a *sentinel* and show how this sentinel can turn the *doubly linked list L* in question 3.c) into a *circular, doubly linked list with* a *sentinel.* **[4 marks]**

4.

a. Explain the concept of *computational complexity* of algorithms. **[3 marks]**

b. Show how *big-O* notation can be used to estimate the following: **[12 marks]**

i. *A constant function*

ii. *A polynomial* $f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$,

where $a_0, a_1, \ldots, a_{n-1}, a_n$ are real numbers.

iii. *F(N)=1+2+3+.....N where N is a positive integer.*

iv. *F(N) = N! where N! is the factorial of a positive integer and F(0)=1.*

v. *F(N)= log(N!) where N is a positive integer.*

vi. *The codes shown below:*

```
int i,j,m,n;

for(i=0;i<m;i++)
        for(j=0;j<n;j++)
printf("%d""%d",&i,&j);
```

c. Using your estimation in 4. b), derive a *big-O* estimate for $f(n)=3n\log(n!)+(n^2+3)\log(n!)$. **[5 marks]**

5. **McCarthy 91** function is defined using the rule $M(n)=\begin{cases} n-10 & \text{if } n>100 \\ M(M(n+11)) & \text{if } n\leq 100 \end{cases}$

for all positive integers n. Write an interactive C++ program to implement a recursive algorithm that determines ***M(n)***. In your program use at least a recursive function *CompCarthy* that accepts as input at least one parameter (representing the value of n) and returns the computed value of ***M(n)***. Your program should estimate the time, in milliseconds, that it takes the CPU to compute the value of ***M(n)***.. **[15 marks]**

6. Write a C++ programme to implement the following:
   a. *Linear search* that searches for the position of an element in the following list {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19}. Your program should allow a user to select a positive integer from the list and then displays the position of that positive integer in the list. **[10 marks]**
   b. *Binary search*, given list A = [1,3,5,7,11,13,17,19,23, 31,33,37]., show how the algorithms will locate the element 19 in the list. **[10 marks]**
   c. Briefly describe the ***divide-and-conquer*** strategy as a problem solving technique. **[5 marks]**