

# Computer Science Foundation Exam

May 2, 2003

## COMPUTER SCIENCE I

### Section I A

**No Calculators!**

KEY

Name: \_\_\_\_\_

SSN: \_\_\_\_\_

Score:

/50

In this section of the exam, there are three (3) problems

You must do all of them.

The weight of each problem in this section is indicated with the problem. The algorithms in this exam are written in C programming language notation. Partial credit cannot be given unless all work is shown.

As always, be complete, yet concise, and above all be neat. Credit cannot be given when your results are unreadable.

**1. (20 points)**

What is the exact output from the following program?

```
#include<stdio.h>
int f1(int a[], int *b, int c, int d)
{
    a[*b] = d;
    a[c] = 4;
    c = *b;
    *b = *b/2 + d;
    d = d + 1;
    return *b + c + d;
}

int main()
{
    int a[] = {6,3,2,5,4,7,9};
    int b, c, z, i;

    b = 4;
    c = 1;
    z = f1(a, &b, a[0], c);
    printf("%d : %d : %d \n", z, b, c);
    for (i = 0; i < 7; i++)
        printf("%d ", a[i]);
    printf("\n");

    b = 5;
    c = 4;
    z = f1(a, &a[1], b, a[5]);
    printf("%d : %d : %d \n", z, b, c);
    for (i = 0; i < 7; i++)
        printf("%d ", a[i]);
    printf("\n");
}
```

Output :

<b>9 : 3 : 1</b>						
<b>6</b>	<b>3</b>	<b>2</b>	<b>5</b>	<b>1</b>	<b>7</b>	<b>4</b>
<b>19 : 5 : 4</b>						
<b>6</b>	<b>8</b>	<b>2</b>	<b>7</b>	<b>1</b>	<b>4</b>	<b>4</b>

**2. (15 points – 6pts(a), 9pts(b))**

(a) Give the contents of the stack at the reference points marked in the expression as A, B and C, as the following postfix expression is evaluated. Write the final value of the expression in the box below.

5 9 \* 64 7 <sup>A</sup> 4 6 + \* 2 8 5 \* <sup>B</sup> - + / <sup>C</sup> \*

7
64
45

A

40
2
70
64
45

B

2
45

C

Final value of the expression is:

<b>90</b>
-----------

(b) Assume that you have a stack S, a queue Q, and the standard stack - queue operations: push, pop, enqueue and dequeue. Assume that print is a function that prints the value of its argument. Execute, in top-to-bottom order, the operations below and answer the following questions.

```

push(S, 5);
enqueue(Q, 4);
push(S, 8);
enqueue(Q, 6);
enqueue(Q, pop(S));
print(dequeue(Q));
enqueue(Q, 7);
push(S, 3);
push(S, dequeue(Q));
print(pop(S));
enqueue(Q, pop(S));
push(S, 2);
enqueue(Q, 5);
push(S, dequeue(Q));
print(dequeue(Q));
print(pop(S));
print(dequeue(Q));
    
```

b.1) Show the output from the print statements (1 pt. each):

4	6	7	8	3
first output	second output	third output	fourth output	fifth output

b.2) After the above operations are completed, how many items are left in stack S?

Answer (2pts):   2  

b.3) After the above operations are completed, how many items are left in queue Q?

Answer (2 pts):   1

**3. (15 points – 5 points each)**

Answer each of the following "timing" questions concerning an algorithm of a particular order and a data set of a particular size. Assume that the run time is affected only by the size of the data set and not its composition and that  $n$  is an arbitrary integer.

- (a) Assume that you are given an algorithm that runs in  $O(N \log_2 N)$  time. Suppose it runs in 20ms for an input size of 16. Estimate the running time of the algorithm for an input size of 64.

$$\frac{16 \log 16}{20} = \frac{64 \log 64}{x} \Rightarrow x = 120$$

Answer: 120ms

- (b) For an  $O(N^k)$  algorithm, where  $k$  is a positive integer, an instance of size  $M$  takes 32 seconds to run. Suppose you run an instance of size  $2M$  and find that it takes 512 seconds to run. What is the value of  $k$ ?

$$\frac{M^k}{32} = \frac{(2M)^k}{512} \Rightarrow 2^k = \frac{512}{32} \Rightarrow 2^k = 16$$
$$\Rightarrow k = 4$$

Answer: 4

- c) What is the computational complexity of the following function?

```
int Mystery(int n)
{
    int i, j, sum= 0;

    for(i=0; i < 10; i++){
        for(j=0; j < i; j++) {
            sum += j * n;
        }
    }
    return sum;
}
```

Answer:  $O(1)$