

Computer Science Foundation Exam

August 1, 2003

COMPUTER SCIENCE I

Section I A

SOLUTION 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] = c;
    a[d] = 3;
    c = *b;
    *b = *b/2 + d;
    d = d + 1;
    return *b + c + d;
}

int main()
{
    int a[] = {8,4,6,2,8,7,9};
    int b, c, z, i;

    c = 6;
    b = 3;
    z = f1(a, &c, a[4], b);
    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[4], c, a[3]);
    printf("%d : %d : %d \n", z, b, c);
    for (i = 0; i < 7; i++)
        printf("%d ", a[i]);
    printf("\n");
}
```

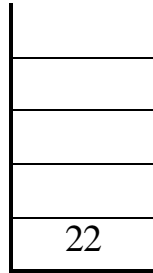
Output :

16			3		6	
8	4	6	3	8	7	8
19			5		4	
8	4	6	3	7	7	8

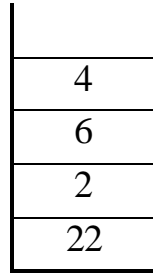
2. (15 points – 7pts(a), 8pts(b))

(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.

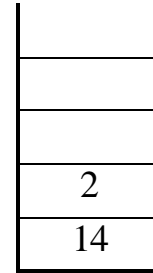
4 6 3 * + ^A 8 2 2 * / 6 4 + ^B - + ^C 2 4 + -



A



B



C

Final value of the expression is:

8

(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, 'T');
enqueue(Q, 'I');
push(S, dequeue(Q));
enqueue(Q, 'I');
enqueue(Q, 'G');
print(dequeue(Q));
enqueue(Q, T);
push(S, 'I');
push(S, dequeue(Q));
print(pop(S));
enqueue(Q, pop(S));
push(S, 'O');
print(pop(S));
enqueue(Q, 'O');
print(dequeue(Q));
enqueue(Q, pop(S));
push(S, dequeue(Q));
print(pop(S));
print(pop(S));
    
```

b.1) Show the output from the print statements:

I	G	O	T	I	T
first output	second output	third output	fourth output	fifth output	sixth output

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

Answer: **0**

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

Answer: **2**

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 $O(N^3)$ time. Suppose it runs in 3ms for an input size of 5. Estimate the running time of the algorithm if the input size is doubled.

24 ms

- (b) Assume that you use an $O(2^n)$ algorithm; to solve a problem instance of size $n = 7$ requires 96 seconds to solve. If your friend used a different sized problem instance and the same algorithm required 12 seconds to solve the problem, what was the size of their problem instance?

Instance size $n = 4$

- (c) What is the computational complexity (in terms of Big-Oh) of the following function?

```
x = 0;
for (i = 1; i <= (2*n); i++)
    for (j = 1; j <= n; j++)
        if (j < i)
            x = x + 1;
```

$O(n^2)$