COP 3502 – Exam #1 – Spring 2004

NAME: KEY

February 10, 2004 (**100 points)**

READ THIS! READ THIS! READ THIS!

- No calculators may be used on this exam.
- Place your answers in the boxes. Use other blank areas to shown your work.
- Show all of your work on every problem.

1. (12 points – 2 pts/answer)

What is the exact output from the following program?

```
#include <stdio.h>

int  q1 (int *, int, int *);

int main ( ) {
     int a= 4;
     int b = 17;
     int *c;

     c = &b;
     a = q1(&a, b, c);
     printf("2. a = %d, b = %d, *c = %d\n", a, b, *c);
     return 0;
}

int q1 (int *x, int y, int *z){
     int a = 5;
     int *p;

     printf("1. *x = %d, y = %d, *z = %d\n", *x, y, *z);
     p = z;
     *p = a + *x;
     return (*x + *z + y);
}
```

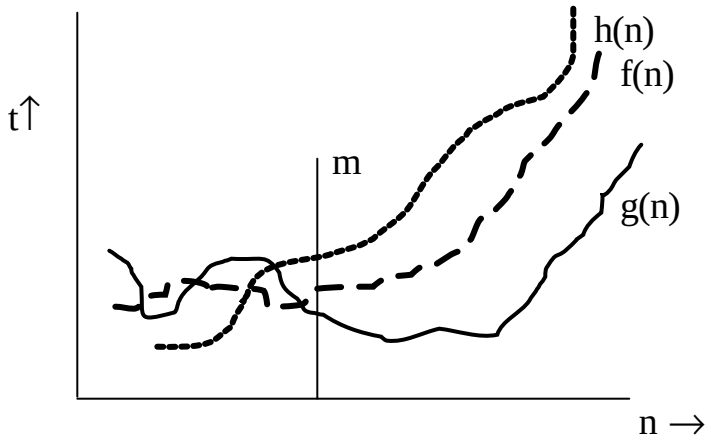
Output is:

1. *x = 4, y = 17, *z = 17

2. a = 30, b = 9, *c = 9

2. (6 points – 2 points each)

Given the figure shown below answer questions (a) through (c).



(a) Using asymptotic notation, describe how h(n) bounds f(n).

f(n) = O(h(n))

(b) Using asymptotic notation, describe how g(n) bounds f(n).

f(n) = Ω(g(n))

(c) Describe the meaning of the value *m* with respect to instance size.

***m* defines the point at which *n* becomes "sufficiently large"**

3. (12 points – 3 points each answer)

Given the following algorithm written in C, fill in the table below to show the results produced for the given input.

```
int  x, y;
scanf ("%d%d", &x, &y)
do {
        y = y + 3;
        if (( x >= y) && (x >= 12))
            x = x – 2;
} while ((x >= y) && (x >= 12));
printf("%d %d\n", x, y);
```

Value of x as input by user	Value of y as input by user	Value of x printed by *printf* statement	Value of y printed by *printf* statement
16	3	**10**	**12**
15	1	**11**	**7**

4. (16 points – 4 points each)

Answer each of the following questions. Be sure to show ALL of your work.

a. What is the decimal equivalent to the binary number: 10011101

answer:

157 since,
$(1\times2^7) + (1\times2^4) + (1\times2^3) + (1\times2^2) + (1\times2^0)$
= 128 + 16 + 8 + 4 + 1
= 157

b. What is the binary equivalent to the decimal number: 79

answer:

1001111 since,
79/2 = 39 remainder 1 (LSB)
39/2 = 19 remainder 1
19/2 = 9 remainder 1
9/2 = 4 remainder 1
4/2 = 2 remainder 0
2/2 = 1 remainder 0
1/2 = 0 remainder 1 (MSB)
reading from MSB to LSB (bottom to top)
gives 1001111

c. What is the octal equivalent of the hexadecimal number: 83AC

answer:

101654 since,
83AC = 1 000 001 110 101 100
= 101654

d. What is the hexadecimal equivalent of the decimal number: 435

1B3 since,
435/16 = 27 with remainder 3
27/16 = 1 with remainder 11 = B
1/16 = 0 with remainder 1
reading from MSD to LSD (bottom to top)
gives 1B3

5. (8 points)

Re-write the program fragment shown below using nested if statements without compound conditions.

```
if ( (a < 0) && (c > 0) ) || (b > 0 )
    printf (“Option #1.\n”);
else
    printf (“Option #2. \n”);
```

Answer:

```
if (b > 0)
   printf (“Option one.\n”);
else  if (a > 0)
       if  (c > 0)
           printf (“Option one.\n”);
       else
           printf (“Option two.\n);
else
    printf (“Option two.\n”);
```

a	c	b	exp
T	T	T	T
T	T	F	T
T	F	T	T
T	F	F	F
F	T	T	T
F	T	F	F
F	F	T	T
F	F	F	F

Truth table for answer

6. (16 points – 4 points each)

Solve each of the following summations. Show ALL your work (use the back if necessary).

(a) $\sum_{i=1}^{37} 7i$

$$7\sum_{i=1}^{37} i = \frac{7n(n+1)}{2} = \frac{7(37)(38)}{2}$$

$$= \frac{7(1406)}{2} = \frac{9842}{2} = 4921$$

(b) $\sum_{k=0}^{n} (4k-1)$

$$4\sum_{k=0}^{n} k - \sum_{k=0}^{n} 1 = \frac{4n(n+1)}{2} - (n+1)$$

$$\frac{4n(n+1)-2(n+1)}{2} = \frac{4n^2+4n-2n-2}{2} = 2n^2+n-1$$

(c) $\sum_{k=0}^{n} (4n)$

$$\sum_{k=0}^{n} (4n) = 4n\sum_{k=0}^{n} 1 = 4n(n+1) = 4n^2+4n$$

(d) $\sum_{n=1}^{50} \frac{3}{4}$

$$\sum_{n=1}^{50} \frac{3}{4} = \frac{3}{4}\sum_{n=1}^{50} 1 = \tfrac{3}{4}(50) = \frac{150}{4} = 37.5$$

7. (10 points – 5 points each)
Answer each of the following questions. Be sure to show ALL of your work.

(a) Form the 2's complement of the binary number using 10 bits: **1101100100**

0010011100

(b) Perform the following subtraction using 2's complement and addition.

```
   11001001
 - 00110101
 ----------
```

2's complement of 00110101 is 11001011

```
      1    1   1 1  carry
   11001001
 + 11001011
 ----------
  10010100
```

8. (6 points – 2 points each)
Given the following algorithm, answer the questions below. Each question refers to replacing the **<conditional expression>** inside the loop condition with some comparison.

```
int x = 1;
while (<conditional expression>)
{
    printf ("%d\n", x);
    x  = x + 2;
}
```

(a) Circle the conditional expression that would cause nothing to be printed by the loop.

x != 1 **x > 0** **x <= 1** **x == 0** **x >= 1**

(b) Circle the conditional expression that would cause the value of **x** to be printed only once.

x > 0 **x < 1** **x == 1** **x >= 1** **x != 1**

(c) Circle the conditional expression that would cause the value of **x** to be printed exactly 5 times.

x >= 10 **x >= 9** **x > 1** **x <= 10** **x != 10**

9. (14 points)

Write a function in C that will compute the value of the following series for given values of *n* and *x*. Do **not** use any C library functions such as pow( ).

$$f(n,x) = \frac{4}{x} + \frac{4(2)}{x^2} + \frac{4(3)}{x^3} + \frac{4(4)}{x^4} + \cdots + \frac{4(n-1)}{x^{(n-1)}} + \frac{4(n)}{x^n}$$

Use the following function header:

```
//Function:  q10: computes f(n, x) as described above.
//Inputs:  x, a double and n, an int.  Restriction: n >= 1 (externally verified).
//Output:  the summation of the first n terms of the series shown above.
double q10 (int n, double x)
```

```
double q10 (int n, double x)
{
    int i;
   double sum = 0; product = 1;
   for (i = 1; i <= n; i++) {
          product = product * x;
          sum = sum + (4 * i)/product;
   }
  return sum;
}
```