Student ID _____ Name _____

cs5f ___ Signature _____

CSE 5A
Midterm
Fall 2006

Page 1 _____ (12 points)

Page 2 _____ (18 points)

Page 3 _____ (34 points)

Page 4 _____ (21 points)

Total _____ (85 points = 81 points + 4 points EC)

This exam is to be taken **by yourself** with only your 1-sided notes, no electronic devices.

Operator Precedence Table

Operators					Associativity
- (unary)	++	--	!		right to left
*	/	%			left to right
+	-				left to right
<	<=	>	>=		left to right
==	!=				left to right
&&					left to right
||					left to right
=	+=	-=	*=	/=	right to left

1. Using the operator precedence table above, evaluate each expression and state what gets printed.

```
int x;
int a = 12;
int b = 7;

x = a % b * 3 - b + a;
printf( "%d\n", x );
```

(3 pts)

```
int x;
int a = 12;
int b = 7;

x = b / 2 + a - 2 * b;
printf( "%d\n", x );
```

(3 pts)

2. What gets printed in the following blocks of statements?

```
int a = 5;
int b = 8;
int c = 15;

if ( (b < a) || (b != 8) && (c == a+10) )
   printf( "True" );
else
   printf( "False" );
```

(3 pts)

```
int x = 3;
int y = 2;
int z = x - 9;

if ( !(z < 6) && (x <= y) || (z < x) )
   printf( "True" );
else
   printf( "False" );
```

(3 pts)

3. Which of the following <u>are not</u> valid C identifiers? (Circle your answer(s).) (3 pts max)
[-1 for each incorrect]

```
Sex-Wax                 cul8r                       double

character               24_Hour_Fitness             Double
```

4. Fill in the blanks with the appropriate types and format specifiers to output the values correctly. (6 pts)

```
void
main( void )
{
    ___________ a = '2';

    ___________ b = 151;

    ___________ c = 4.20;

    printf( "b = %____\nc = %____\na = %____\n", b, c, a );
}
```

5. What gets printed? (9 pts)

```
void
main( void )
{
    int num = 3;

    switch ( num = num + 2 )
    {
        case 2:
                printf( "A\n" );
                num = num + 2;

        case 7:
                printf( "B\n" );
                num = num + 4;
                break;

        case 3:
                printf( "C\n" );
                num = num + num;
                break;

        case 5:
                printf( "D\n" );
                num = num + 5;

        default:
                printf( "E\n" );
                num = num + 8;
                break;
    }

    printf( "num = %d\n", num );
}
```

6. Write an equivalent **while loop** for the following **for loop**. (12 pts)

```
for (i = -12; i <= j; ++i )
{
    printf( "%d %d\n", i, j );
    j = i + j;
}
```

Equivalent **while loop**

7. What gets printed in the following block of statements? (8 pts)

```
#define SIZE 8

int i;
int array[SIZE] = { 2, 7, -4, 5, 16, 0, -2, 4 };

for ( i = 0; i < SIZE; ++i )
    if ( array[i] >= 4 )
        printf( "%d\n", array[i] );
```

8. (14 pts)

```
#include <stdio.h>
#define SIZE 8
void function1( int var1, int var2 );

int
main( void )
{
    int a[SIZE] = { 2, 3, -4, 5, 16, 0, -2, 4 };
    int z;

    scanf( "%d", &z );     /* Read an integer as input */

    if ( a[z] > a[z-1] )
    {
        z = 4;
        function1( a[z], z );
    } else {
        z = 1;
        function1( z, a[z] );
    }

    return 0;
}

void
function1( int var1, int var2 )
{
    int i = 0;

    do {
        printf( "%d\n", var1 );
        ++var1;
        ++i;
    } while ( i < var2 );
}
```

What gets printed if the input is **4**?

What gets printed if the input is **2**?

9. What gets printed? (16 pts)

```
 1      #include <stdio.h>
 2
 3      #define SIZE 7
 4
 5      int jenny( int x );
 6
 7      int
 8      main( void )
 9      {
10          int array[SIZE];
11          int i;
12
13          for ( i = 0; i < SIZE; ++i )
14          {
15              array[i] = jenny( i );
16          }
17
18          for ( i = 0; i < SIZE; ++i )
19          {
20              printf( "%d\n", array[i] );
21          }
22
23          printf( "i = %d\n", i );
24
25          return 0;
26      }
27
28      int
29      jenny( int x )
30      {
31          int number[] = { 3, 0, 9, 8, 6, 7, 5 };
32
33          if ( (x + 3) < SIZE )
34              return ( number[x + 3] );
35          else
36              return ( number[(x + 3) - SIZE] );
37      }
```

Use the following to answer the questions below: (5 pts)

A) Return Type
B) Formal Parameter
C) Function Prototype (Function Declaration)
D) Actual Argument
E) Local Variable
F) Function Definition
G) C Preprocessor Directive

What is `x` in line 29? _____

What is `number` in line 31? _____

What is the `#define` in line 3? _____

What is the `int` in line 7? _____

What is line 5? _____

Scratch Paper