

Student ID _____

CSE 5A Midterm Fall 2004

Name _____

Signature _____

cs5a _____

Page 1 (12) _____

Page 2 (19) _____

Page 3 (32) _____

Page 4 (21) _____

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

Total (84) _____

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 = 11;
int b = 4;
```

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

(3 pts)

```
int x;
int a = 11;
int b = 4;
```

```
x = a / 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 == b+10) )
    printf( "True" );
else
    printf( "False" );
```

(3 pts)

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

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

(3 pts)

3. Which of the following are not valid C identifiers? (Circle your answer(s).) (4 pts max)
[0 if all circled; -? for each incorrect]

FLOAT	A_PhD_In_Swing	2_Degrees_In_BeBop
A_Master's_In_Rhythm	int	A.F.I.

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

```
void
main( void )
{
    _____ a = 98.6;
    _____ b = '9';
    _____ c = 420;

    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 5:
            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 3:
            printf( "D\n" );
            num = num + 5;

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

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



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

```
i = 21;
while ( i >= j )
{
    j = i / j;
    printf( "%d %d\n", i, j );
    --i;
}
```

Equivalent for loop

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

```
#define SIZE 8

int i;
int array[SIZE] = { 11, 7, 3, 4, 6, 2, -1, 5 };

for ( i = 0; i < SIZE; ++i )
    if ( i % 2 == 1 )
        printf( "%d\n", array[i] );
```

8. (12 pts)

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

int
main( void )
{
    int a[SIZE] = { 11, 7, 3, 7, 6, 2, -1, 3 };
    int z;

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

    if ( a[z] < a[z+1] )
    {
        z = 3;
        function1( a[z], z );
    } else {
        z = 7;
        function1( z, a[z] );
    }

    return 0;
}

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

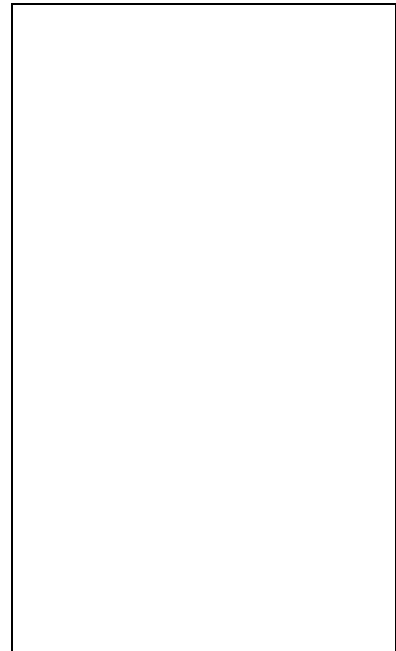
    do {
        printf( "%d", var1 );
        ++var1;
        ++i;
    } while ( i < var2 );
}
```

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 Janet( 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] = Janet( 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    Janet( int x )
30    {
31       int number[SIZE] = { 15, 23, 34, 42, 50, 67, 76 };
32
33       if ( x < SIZE / 2 )
34           return ( number[x] / 10 );
35       else
36           return ( number[x] % 10 );
37    }
```



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

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

What is array in line 10? _____

What is line 5? _____

What is the #include in line 1? _____

What is x in line 29? _____

What is the int in line 28? _____

Scratch Paper