

Exam 1 Review

COP 3223 Section 1

1) Which of the following prints out exactly 5
backslashes?

A)printf("\\\\\\\\");

B)printf("\\\\\\\\\\");

C)printf("\\\\\\\\\\\\\\\\\\");

D)printf("\\n\\n\\n\\n\\n");

E)None of the Above

2) What is the value of the following arithmetic expression?

$$2+3*4 - 25/(2+3)$$

A)-2

B)-1

C)15

D)20

E)None of the Above

3) What is the value of the following arithmetic expression?

$$17 - 77\%9 + 2*(3 - 8\%65)$$

A)-1

B)0

C)2

D)16

E)None of the Above

4) What is the output of the following segment of code?

```
int a=6, b=2;  
if (a > 3)  
    printf("A");  
if (b > a)  
    printf("B");  
else  
    printf("C");
```

- A)A B)B C)C D)AB
E)None of the Above

5) What is the output of the following segment of code?

```
int a=4;  
if (a > 3) {  
    printf("A");  
}  
else  
    printf("B");  
    printf("C");
```

- A)A B)C C)AC D)BC
E)None of the Above

6) What is the value of sum at the end of the following code segment?

```
int sum=0, i;  
for (i=1; i<13; i=i+2)  
    sum = sum+i;
```

- A)13 **B)36** C)49 D)78
E)None of the Above

7) How many times does the body of the while loop below run?

```
int n = 20;  
while (n > 0) {  
    printf("%d\n", n);  
    n = n/2;  
}
```

- A)4 B)6 C)10 D)20
E)None of the Above

8) What is the output of the following segment of code?

```
int a = 3;  
do {  
    a = a*2;  
    printf("%d ", a);  
} while (a < 50);
```

- A) 3 6 12 24 48 B) 3 6 12 24 C) 6 12 24 48 96
D) 6 12 24 48 E) None of the Above

9) How many times does, "Go UCF!" get printed in the following code segment?

```
int i;  
for (i=0; i<100; i++) {  
    if (i%10 > 6)  
        continue;  
    printf("Go UCF!\n");  
}
```

- A) 60 B) 70 C) 90 D) 100
E) None of the Above

10) Which of the following is a good reason to include comments in your code?

- A) They make the code compile faster.
- B) They make the code run faster.
- C) They increase your frame rate to 60 frames/second.
- D) They make it easier for someone else reading your code to understand it.
- E) None of the Above

5) What is the output of the following segment of code?

```
int a=8, b=5;  
if (a = 3)  
    printf("A");  
b = 2*a;  
printf("%d", b);
```

- A) 6
- B) 10
- C) A6
- D) A10
- E) None of the Above

6) What is the output of the following segment of code?

```
int a=2, b=9;  
if (a > 3);  
    printf("A");  
if (b > 8)  
    printf("B");  
else  
    printf("C");
```

- A)A
- B)B
- C)AB
- D)AC
- E)None of the Above

10) How many times does, "Go UCF!" get printed in the following code segment?

```
int i;  
for (i=0; i<100; i++) {  
    if ((4*i+8) % 100 == 0)  
        break;  
    printf("Go UCF!\n");  
}
```

- A)23
- B)98
- C)99
- D)100
- E)None of the Above

- 1) (15 pts) The following table shows telephone area codes in the state of Georgia along with the largest area in each city.

<i>Area Code</i>	<i>Major City</i>
229	Albany
404	Atlanta
470	Atlanta
478	Macon
678	Atlanta
706	Columbus
762	Columbus
770	Atlanta
912	Savannah

Write a switch statement whose controlling expression is the variable `area_code`. If the value of `area_code` is in the table, the switch statement will print the corresponding city name. Otherwise, the switch statement will print "Area code not recognized". Make the switch statement as simple and efficient as possible.

```
switch ( area_code ) {
    case 229:
        printf("Albany");
        break;
    case 404:
    case 470:
    case 678:
    case 770:
        printf("Atlanta");
        break;
    case 478:
        printf("Macon");
        break;
    case 706:
    case 762:
        printf("Columbus");
        break;
    case 912:
        printf("Savannah");
        break;
    default:
        printf("Area code not recognized");
}
```

- 2) (15 pts) Complete the program below so that it reads in a list of grades entered by the user (terminated by -1) and prints out both the minimum and maximum of those grades. Note: All grades entered are guaranteed to be in between 0 and 100, inclusive. The last value entered, -1, is NOT a grade, but simply signifies the end of the list.

```
#include <stdio.h>
int main(void) {

    int min=101, max=-1, grade;
    printf("Please enter all of your grades, followed by -1.\n");
    scanf("%d", &grade);

    while ( grade != -1 ) {
        if ( grade > max ) max = grade;
        if ( grade < min ) min = grade;
        scanf("%d", &grade);
    }

    printf("The minimum grade was %d.\n", min);
    printf("The maximum grade was %d.\n", max);
    return 0;
}
```

- 3) (20 pts) The mathematical constant e can be expressed as an infinite series

$$e = 1 + 1/1! + 1/2! + 1/3! + \dots$$

Write a program that approximates e by computing the value of

$$1 + 1/1! + 1/2! + 1/3! + \dots + 1/n!$$

where n is an integer entered by the user. Don't concern yourself with n getting too large to fit in an int type. HINT: The factorial of an integer n is simply the product of all positive integers less than or equal to n . Thus, $3! = 1 \times 2 \times 3$. Also note that $0! = 1$

```
#include <stdio.h>
#include <stdlib.h>

int main( void ) {
    int n;
    printf("Enter integer corresponding to degree of e approximation: ");
    scanf("%d", &n);
    int current_factorial = 1;
    float e_approximation = 1.0;
    int index;
    for ( index = 1; index <= n; index++ ) {
        current_factorial *= index;
        e_approximation += ( 1 / (float) current_factorial );
    }
    printf("e is approximatly %f\n", e_approximation);
    getchar(); //system("pause");
    return 0;
}
```