

Final Exam – Fall 2021

COMP – 1411 Computer Programming I

Total Points: 50

Time: 3 Hours

(Submit your answers in a single .pdf file. Use the link created for final exam on D2L to upload your answers.)

Note: You are not allowed to use compiler/editor. You do not need to write comment.

1. [4 points] Consider the following program segment. What will be the output generated by the **printf** statements?

```
//global variables
int x = 1;
int y = 2;

//function definition
int fun(int z ){
    .....
    .....
    int x = 5;
    z = 2;
    y = 7;
    printf("Inside fun: x = %d\n", x);
    printf("Inside fun: y = %d\n", y);
    z = foo( x);
    .....
    .....
    return x;
}

//function definition
int foo(int w){
    .....
    .....
    w = 5;
    printf("Inside foo: x = %d\n", x);
    printf("Inside foo: y = %d\n", y);
    .....
    .....
    return w;
}
```

2. [3 points] Consider the following program segment. What will be the output generated by the **printf** statements?

```
int foo(){
    .....
    .....
    int x = fun(5);
    x = fun( 6);
    return x;
}

int fun(int x){
    int y = 1;
    printf("Inside fun: y = %d\n", y);
    y += 1;
    ++x;
    return y;
}
```

3. [6 points] Consider the following program segment. What will be the output generated by the **printf** statements?

```
int foo(){
    int A[ ] = {2,4,6,8};
    int x = 1;
    int n = 4;
    printf("Inside foo : Before fun\n");
    printf("x = %d\n", x);
    for(int i = 0; i < n; i++)
        printf("%d ", A[i]);
    printf("\n");
    int z = fun(A, n, x);
    printf("Inside foo : After fun\n");
    printf("x = %d\n", x);
    for(int i = 0; i < n; i++)
        printf("%d ", A[i]);
    printf("\n");
}
```

```

int fun(int A[], int n, int x){
    x++;
    for(int i = 0; i < n; i++)
        A[i]++;
    printf("Inside fun: \n");
    printf("x = %d\n", x);
    for(int i = 0; i < n; i++)
        printf("%d ", A[i]);
    printf("\n");
    return n;
}

```

4. [3 points] Write True or False for each of the following statements in the context of C programming language.

- a) A two-dimensional array can store values of two different data types in two dimensions.
- b) The following is a valid string initialization in C:

```
char messi[14];
messi = "Leo Golden 07";
```
- c) You are not allowed to pass a struct variable to a function by value. Struct variables must be passed to a function by reference.

5. [2 points] Find and correct error (if any) in the following code segment.

```

.....
.....

float x = 5.8;
int y = 3;
int *p = &x;
int *q = &y;
printf("x = %f\n", *p);
printf("y = %d\n", y);
.....
.....

```

6. [8 points] Consider an integer array `arr[] = {-5,-4,-3,-2,-1,10}`. In the following table 'Index' represents indexes of

the array `arr[]`, 'Value' represents the values of the elements of the array, and the 'address' represents the memory addresses of the starting byte of the array elements.

Index	0	1	2	3	4	5
Value	-5	-4	-3	-2	-1	10
Address	100	104	108	112	116	120

Consider the table and the description as mentioned above and find the output of the **printf** statements from the following program segment.

```
int arr[ ] = {-5, -4, -3, -2, -1, 10};
int *p = arr;
++*p;
int **q = &p;
printf("*p = %d\n", *p);
printf("p = %d\n", p);
printf("arr = %d\n", arr);
p += 3;
printf("*p = %d\n", *p);
printf("p = %d\n", p);
printf("arr = %d\n", arr);
printf("**q = %d\n", **q);
for(int i = 0; i < 6; i++)
printf("%d ", arr[i]);
```

7. [9 points] Suppose you have an input data file, `input.txt`, stored in the local directory. The file contains players' first name (space) last name (space) number of Ballon d'Or won.

```
Lionel Messi 7
Cristiano Ronaldo 5
Michel Platini 3
.....
```

`input.txt`

Suppose a struct is defined as follows:

```

struct BallonDor{
char name[50];
int numBorWon;
};

```

- a) Write C codes to declare an array of struct of **BallonDor** type.
 - b) Write C codes to read the entire content of the file. This means read the strings (names) and the integers (number of Ballon d'Or won) from input.txt and store to the corresponding members of the structure variable (array of struct). Show the declaration and the use of pointer variables for handling the input file.
 - c) Write C code to store the strings (names) and the integers (number of Ballon d'Or won) to a file output.txt. In other words, write (store) the array of struct to the output file. Show the declaration and the use of pointer variables for handling the output file.
8. [5 points] Write a function definition, **countWord**, that has only one parameter which is a line of text as a character array (name the array as **sentence**) and returns the number of words in the **sentence**.
9. [10 points] Consider a 2-D array, **result**. This array stores student IDs of N ($1 \leq N \leq 60$) students and marks obtained by each student in M ($1 \leq M \leq 3$) number of exams.

Let's assume that students' IDs are 4 digits long. For example, the 2-D array, **result**, can be the following:

1111	20	33	40
1123	17	26	31
1131	15	20	42
1145	22	35	41
1151	20	27	39

The values at the left-most column represent student IDs, and the values of the other columns are the marks. Each row is a record for one student. For example, the first row represents the marks (20, 33, 40) of the student with ID 1111.

Write the definitions of the following user-defined functions. You may use other parameters (arguments) in the user-defined functions as required.

topScorer - this function takes the 2-D array, **result**, as a function argument, and returns the ID of the top scorer. For example, if you consider the above mentioned 2-D array, this function will return 1145.

searchStudent - this function receives the 2-D array, **result**, and an ID that a user looks for. This function searches for the ID (target ID) and returns the total marks (marks in all exams) obtained by the target student. Implement this function as a void function. Utilize the concept of reference parameters using pointers for implicitly returning the value of total marks. For simplicity, let's assume that the user inputted ID always exist in the 2-D array.