Princess Sumaya جامعة الأميرة سميّة للتكنولوجيا University for Technology

The King Hussein School for Computing Sciences
Department of Computer Science
Structured Programming - Spring 2022

Second Exam

Full Name: **Student ID:**

Question	Points	Score
1	4	
2	3	
3	4+1	
4	7	
5	7	
Total	25+1	

Circle your section:

- ○ Dr. Mu'awya Al-Dala'ien (section 1)
- ○ Dr. Rawan Ghnemat (section 2)
- ○ Dr. Abdullah Aref (section 3)
- ○ Dr. Mu'awya Al-Dala'ien (section 4)
- ○ Dr. Rawan Ghnemat (section 5)
- ○ Dr. Sawsan Alshatnawi (section 6)
- ○ Dr. Mohammad Al Nabhan (section 7)
- ○ Dr. Sawsan Alshatnawi (section 8)
- ○ Dr. Mohammad Abu Snober (section 9)
- ○ Dr. Mohammad Abu Snober (section 10)
- ○ Dr. Khaled Mansour (section 11)
- ○ Dr. Abedalrhman Alkhateeb (section 13)
- ○ Dr. Khaled Mansour (section 14)
- ○ Dr. Rafat Hammad (section 15)

Question 1 (4 points)

Fill the **Output** column in the table below with the output of the code provided in the **Code** column. If the code does not compile, write "**compilation error**" instead of the output.

	Code	Output
1.	`int a[2][2] = {{1, 2}, {3, 4}};` `printf("%d", a[0][1]);`	
2.	`int a[3][3] = {{1, 2}, {3, 4}};` `printf("%d", a[2][2]);`	
3.	`int a[][] = {{1, 2}, {3, 4}};` `printf("%d", a[1][1]);`	

4.	`int x = 2;`
	`do printf("%d ", x--);`
	`while (x >= 2);`	

5.	`for (int i = 0; i < 3; i++)`
	`  if (i == 1) continue;`
	`  else  printf("%d ", i);`	

6.	`for (int i = 0; i < 3; i++)`
	`  if (i == 1) break;`
	`  else  printf("%d ", i);`	

7.	`for (int i = 0; i < 2; i++)`
	`  printf("%d ", i);`
	`for (int j = 0; j < 2; j++)`
	`  printf("%d ", j);`	

8.	`void f(int x) {`
	`  if (x == 3) break;`
	`  else  printf("Hello");`
	`}`
	
	`int main() {`
	`  f(3);`
	`  return 0;`
	`}`	

Question 2 (3 points)

Convert the following function to a recursive function:

```
void boom(int n) {
    while (n > 0)
        printf("%d ", n--);
    printf("Boooom!");
}
```

Question 3 (4+1 points)

PART 1.

```
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n - 1; j++) {
        if      (a[i][j] == a[i][j+1]) printf("1 ");
        else if (a[i][j] == a[i][j-1]) printf("0 ");
    }
}
```

A. Provide an example of an array `a[][]` of size `[n=3]x[n=3]` that will cause the above code to print **`1 1 1 1 1 1.`**

B. Provide an example of an array `a[][]` of size **`[n=3]x[n=3]`** that will cause the above code to print **`1 1 1 1 1 0.`**

C. [+1 point] Provide an example of an array `a[][]` of size **`[n=3]x[n=3]`** that might cause the above code to crash.

Note. This part is a **bonus** question. Do **not** spend time on it until you are done with the other required questions.

PART 2.

```
void f1(int a[], int n) {
      for (int i = 0; i < n; i++)
            for (int j = 0; j < n - 1; j++) {
                  int hold = a[j];
                  a[j] = a[j+1];
                  a[j+1] = hold;
            }
}
```

D. What are the contents of array **a[]** after calling function **f1** if **n = 2** and **a[] = {2, 1}** ?

E. What are the contents of array **a[]** after calling function **f1** if **n = 100** and **a[] = {100, 99, 98, 97, …, 3, 2, 1}** ?

Question 4 (7 points)

A. [4 points] Implement a function named **sudoku**, which receives as an argument a 2D array of integers of size 9x9. The function returns **1** if every column sums to **45** and every row sums to **45**. The function returns **0** otherwise.

B. [3 point] Write a program that creates a `2D` array of size `9x9`, fills it with random integers between `1` and `9` (inclusive) and then uses function **`sudoku`** to check if every row and every column in the array sums to `45`. If this is true, your program must print `"what a surprise!"`.

Question 5 (7 points)

In Number Theory, a Taxicab Number is a number that can be expressed as a sum of cubes in *more than one way*. For example, `1729`, `4104` and `13832` are taxicab numbers, because:

$$1729 = 1^3 + 12^3 \quad \text{and also} \quad 1729 = 10^3 + 9^3$$
$$4104 = 2^3 + 16^3 \quad \text{and also} \quad 4104 = 9^3 + 15^3$$
$$13832 = 20^3 + 18^3 \quad \text{and also} \quad 13832 = 24^3 + 2^3$$

A. [5 points] Implement a function named **`taxicab`** that receives an integer and prints `"taxicab"` if the integer is a taxicab number and `"not taxicab"` otherwise.

B. [2 points] Reimplement function **`taxicab`** such that it prints all the taxicab numbers that are less than the received integer.