CS1131 (Data Structure and Algorithms)
Quiz-1

B

3

Time - 40 Minutes

Name Kalpana Baghel Roll No. 2023B1ech039

Instructions:-

1. **All programs are compiled and run in Linux GCC compiler.**
2. **All Questions are Compulsory.**
3. **If there is an error in the code, kindly explain the error in one line.**

Q1. Fill in the blanks to make the code run **3 Marks**

```
typedef struct {
      int width;
      int height;
      int area;
} rectangle;
//Calculates the area of the rectangle and puts it into member area
void cal_areas(int *r1.width, int *r1.height
{
 int  r1.area          = *r1.width      *  *r1.height ;
}
int main()
{
     rectangle r1 = {20, 10, 0};
     //function call to calculate the area of r1
     cal_areas(r1.width, r1.width);
     printf("The area of the rectangle r1 is %d\n", r1.area );
     return 0;
}
```

1

Q2. What will be the error/output of the following C code? **1 Mark**

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int i;
    int *ptr = (int *) malloc(5 * sizeof(int));
    for (i=0; i<5; i++)
        ptr[i] = i;
    printf("%d ", *ptr);
    printf("%d ", ++*ptr);
    free(ptr);
}
```

0

X Output: 4 garbage value

Q3. What will be the error/output of the following C code? **1 Mark**

```
int main()
{
   double arr[2] = {20.0, 25.0}, *p, *q;
   p=arr;
   q=p+1;
   printf("%d, %d",(int)(q-p), (int)(*q-*p));
   return 0;
}
```

1/2

output: 8 X

Q4. Declare a self-referential struct of type student having rollnumber, name, and an array of marks.

1 Mark

```
typedef struct {
        int rollnumber;
        char name[20];
        int marks[10];
        student *next;
} student;
```

Q5. Define a type island having members island name, country, population. Create an array of islands. Write a function to return the island with the maximum population. Create the array dynamically and take the size of the array from the user

[4 Marks]

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

typedef struct {
        char name[50];
        char country[50];
        int population;
} island;

int main() {
    int n;
    scanf("%d", &n);
    island *island = (island*) malloc(sizeof(island) * n);
    int maxpopulation = islandmax(island[n].population, n);
    printf("The max population island is %s", island.population[maxpopulation]);
}

int islandmax(int arr[], int n){
    int max = arr[0];
    for(int i=0; i<n; i++){
        if(max < arr[i]){
            max = arr[i];
        }
    }
    return i;
}
```