CS11001: Programming and Data Structures

Class Test II

Total Marks :

Answer All Questions. Write your answers in the boxes provided. Total Marks = 2 x 10 = 20.

Section:	**Roll:**	**Name:**

For Questions 1-5, consider the following structures representing a railway station, a train, and a ticket. In the structure `train`, `n_stops` represents the number of stops of a train including the start and end stations.

```
struct station {
int stn_code;
char stn_name[30];
}
```

```
struct train {
int trn_number;
char trn_name[30];
int n_stops;
struct station stops[100];}
```

```
struct ticket {
int pnr;
struct train trn;
char psgr_name[30];
struct station from;
struct station to;}
```

1. If an `int` is stored using 4 bytes, and a `char` is stored using 1 byte. How many bytes are required to store?

(i) `struct train` _____3438_____

(ii) `struct ticket` _____3536_____

2. Suppose we have the following structure initialization:

```
struct train trn = {15905, "Vivek Express", 2, 7280, "DBRG", 125, "CAPE"};
```

What is the value of `trn.stops[1].stn_name[1]` _____`A'_____

3. Complete the following function to determine if a train `trn` runs between two given stations `from` and `to`.

```
int IsRun(struct train trn, struct station from, struct station to){
int i, j, fflag = 0, tflag = 0;
for(i=0;i < trn.n_stops - 1;i++){

     if (trn.stops[i].stn_code == from.stn_code          ) { fflag = 1;
     for(j = i+1 ; j < trn.n_stops; j++){

     if (trn.stops[j].stn_code == to.stn_code           )   tflag =1;
     }}}
     return (fflag && tflag);}
```

4. A structure array `struct train all_trns[1000]`, stores the data for all the `N` (<1000) trains in Indian Railway. Complete the function below to find out all the trains between `from` and `to` stations and to store them in the input structure array `tbs[ ]`. Use the `IsRun` function defined in Question 3.

```
int trns_btwn_stns(struct station from, struct station to, int N, struct train
all_trns[ ], struct train tbs[ ]){

int t, n=0;

for(t = 0; t < N ; t++){


if (IsRun(all_trns[t], from, to)                              ){


tbs[n] = all_trns[t]                                    ; n++;}

return n;}}
```

5. A structure array `struct ticket sold_tkt[1000]` stores all the `T` (< 1000) tickets sold in a day. Assume, only one person travels in a single ticket and boards the train from the *from* station of the ticket. Also, the designated *train* of a ticket stops at the *from* and *to* stations of the ticket. Complete the function below which takes as input a train `trn`, and the `sold_tkt[]` array, and returns the number of passengers, with tickets, who boarded the train at an intermediate station (i.e., not at the starting station).

```
int boarded_on_way(int T, struct ticket sold_tkt[], struct train trn){

int tk, n = 0;

for(tk=0; tk < T; tk++){

if(sold_tkt[tk].trn.trn_number == trn.trn_number){


if(sold_tkt[tk].from.stn_code != trn.stops[0].stn_code) n++;

return n;

}}}
```

6. How many times is the following recursive function `int g(int n)` invoked to compute `g(5)`?

```
int g(int n){
       if (n==0) return 0;
       else if (n==1) return 1;
       else if (n==2) return 2;
       else return g(n-1)+g(n-2)+g(n-3);}
```

13

7. Convert the recursive program in the left box to an iterative program in the right box.

```
int sum(int n) {
if (n < 1) return 0;
return sum(n - 1) + sum(n - 2) + n;
}
```

```
int sum(int n) {
int i, fp = 0, fc = 0, fn;
for (i = 0; i <= n; i++) {

fn = fc + fp + i          ;

fp = fc                   ;
fc = fn;
}
return fc;
}
```

8. What is printed by the following program?

xzxyzy

```
#include<stdio.h>
void t(int n, char fp, char tp, char ap){
     if(n==1){
     printf("%c%c",fp,tp);
     return;
     }
     t(n-1,fp,ap,tp);
     printf("%c%c",fp,tp);
     t(n-1,ap,tp,fp);
     return;
}
void main(){
     t(2,'x','y','z');
}
```

9. If our universe consists of only the set of nonnegative integers, the even and odd numbers can be characterized as follows: a number is *even* if its predecessor is odd, a number is *odd* if is not even, the number 0 is even by definition. Complete the C functions below which return `1` when the input number *n* is even/odd respectively. Both the functions are defined in the same program and they call each other.

```
int IsEven(unsigned int n) {
if (n == 0) {
return 1;

} else {

return IsOdd(n - 1);
}}
```

```
int IsOdd(unsigned int n) {

return !IsEven(n);

}
```

10. A recursive algorithm may require more computation time and memory than its iterative version.

(A) TRUE (B) FALSE (tick one)

Rough Work