Q1
3.5 Points

If a and b can take on values in the set $\{0, 1, 2, 3\}$, when will the following conditions evaluate to True. Give your answer in terms of (a, b) value pairs. For example, the condition $(a > b)$ will evaluate to True with the value pairs $(1, 0), (2, 0), (3, 0), (2, 1), (3, 1), (3, 2)$.

Q1.1
0.5 Points

(a < b)

(0,1), (0,2), (0,3), (1,2), (1,3), (2,3)

Q1.2
0.5 Points

(a < b) && (b > a)

(0,1), (0,2), (0,3), (1,2), (1,3), (2,3)

Q1.3
0.5 Points

(a < b) || (b > a)

(0,1), (0,2), (0,3), (1,2), (1,3), (2,3)

Q1.4
0.5 Points

(a – b == 1)

(1,0), (2,1), (3,2)

Q1.5
0.5 Points

(a – b == 1) && (a > 1)

(2,1), (3,2)

Q1.6
0.5 Points

(a – b)

(0,1), (0,2), (0,3), (1,0), (1,2), (1,3), (2,0), (2,1), (2,3), (3,0), (3,1), (3,2)

Q1.7
0.5 Points

!(a – b)

(0,0), (1,1), (2,2), (3,3)

Q2 (Spot error and fix)
4 Points

Consider the following programming problem and a C code that attempts to solve it:

- Take a positive integer as input from the user and output the sum of even numbers that are less than or equal to the input integer.

```
#include<stdio.h>
void main(){
    int n, sum;
    int i=1;

    printf("Enter a positive integer:");
    scanf("%d", &n);

    while(i < n){
        if(i%2 == 0)sum+=i;
        i++;
    }
    printf("The answer is %d\n", sum);
}
```

What are the errors in the above program? Note that there may be more than one error.

The condition of the while loop is i<n which is wrong since it does not work for the case when i==n, as n can be even the loop, will not add n in the sum in the case when it is even. Here since we have to find the numbers less than or equal to we have to include n in the loop
Also, the variable sum should be initialized to 0 which can hold a garbage value if uninitialized and hence the initialization should be done explicitly

Make appropriate changes to correct the program.

```
#include<stdio.h>
void main(){
   int n, sum=0;
   int i=1;

   printf("Enter a positive integer:");
   scanf("%d", &n);

   while(i <= n){
      if(i%2 == 0)sum+=i;
      i++;
   }
```

```
    printf("The answer is %d\n", sum);
}
```

Q3 (Spot error and fix)
4 Points

Consider the following programming problem and a C code that attempts to solve it:

- Take a positive integer val as input from the user and output the largest integer m such that $(1 + 2 + 3 + ... + m) \leq val$.

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

    int val;
    int i=1, sum=0;

    printf("Enter positive integers val:");
    scanf("%d", &val);

    while(sum < val){
        sum = sum + i;
        i = i+1;
    }
    printf("The answer is %d\n", i);
}
```

What is the error in the above program?

Here the loop breaks when the value of sum is greater than val in this case i is already greater than the required value since it has been updated in the previous iteration
For example if val is 2 then loop runs twice and i is updated to 2 but the next condition is evaluated after this and 3<2 becomes false but i remains updated to 2 this thus finds the incorrect value of i
We should instead increment i after checking that sum + (i+1) remains less than or equal to val

Make an appropriate change to correct the program.

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

```
    int val;
    int i=1, sum=0;

    printf("Enter positive integers val:");
    scanf("%d", &val);
    sum = sum +i ;
    while(sum + i + 1 <= val){
        i = i+1;
       sum = sum + i;
    }
    printf("The answer is %d\n", i);
}
```

Q4
6 Points

Write a C program that prints the following pattern using * and @:

```
*
*@
*@*□
*@*@
*□
*@
*@*
*@*@□
*@*@*□
*□
*@
*@*
*@*@
*@*@*□
*@*@*@
```

Use nested loops to solve this problem. Do not give a dumb solution with 15 printf statements.

```
#include<stdio.h>

void main(){
```

```
    for(int i = 1 ; i<=3 ; i ++){
        for(int j = 1 ; j<=i+3;j++){
            for(int k = 1 ; k<=j ; k ++ ){
                if(k%2==1){
                    printf("*");
                }else{
                    printf("@");
                }
            }
            printf("\n");
        }
    }
    return ;
}
```

Q5
2.5 Points

Without calculating, give the result of the following operations (in hexadecimal):

Q5.1
0.5 Points

```
0xabcdef * 16
```

0xabcdef0

Q5.2
0.5 Points

```
0xabcdef / 256
```

0xabcd

Q5.3
0.5 Points

```
0xabcdef & 0xf0f0f0
```

0xa0c0e0

Q5.4
0.5 Points

```
0xabcdef | 0xf0f0f0
```

0xfbfdff

Q5.5
0.5 Points

```
~ 0xabcdef
```

0xff543210

Q6
-20 Points

Re-minor/viva

Minor Exam COL100

● Graded

8 Days, 22 Hours Late

Student
Chinmay Mittal

Total Points
19 / 0 pts

Question 1
(no title) **3.5** / 3.5 pts

1.1 (no title) **0.5** / 0.5 pts

1.2 (no title) **0.5** / 0.5 pts

1.3 (no title) **0.5** / 0.5 pts

1.4 (no title) **0.5** / 0.5 pts

1.5 (no title) **0.5** / 0.5 pts

1.6 (no title) **0.5** / 0.5 pts

1.7 (no title) **0.5** / 0.5 pts

Question 2
(Spot error and fix) **4** / 4 pts

Question 3
(Spot error and fix) **3** / 4 pts

Question 4
(no title) **6** / 6 pts

Question 5
(no title) **2.5** / 2.5 pts

5.1	(no title)	**0.5** / 0.5 pts
5.2	(no title)	**0.5** / 0.5 pts
5.3	(no title)	**0.5** / 0.5 pts
5.4	(no title)	**0.5** / 0.5 pts
5.5	(no title)	**0.5** / 0.5 pts

Question 6

(no title) **0** / -20 pts