

ESC101 - QUIZ 2 Session 2

Total points ?

The quiz will be conducted in two sessions of 20 minutes each with a gap of 10 minutes in between two sessions.

Read all instructions mentioned in the problem statement carefully before attempting it and try to keep your answers precise. Make sure to submit your response on time. Auto-submission is not available and if you fail to submit on time, you will get zero marks. If there are any issues like internet/power outage contact your Tutor ASAP.

Problem 4

2 of 4 points

The output of the following code is "30 50". Complete the code so that it does the required task correctly. If a line is to be left empty, write '/**/'.

✓ Line 5-Blank 1

1/1

**



✓ Line 5-Blank2

1/1

/**/



Fill the blanks in the code.

```
1 #include <stdio.h>
2
3 void f(int *a, int ***c){
4
5     _____c = _____a;
6 }
7
8 int main(void) {
9
10    int a[] = {10, 20, 30, 40, 50, 60, 70};
11    char *b;
12    int **c = &b;
13
14    f(a, &c);
15
16    // No access to array a beyond this comment
17
18    printf("%d %d", *( b_____ ), *( _____ b + 2) )
19    ;
20    return 0;
21 }
```

✗ Line 18-Blank20

0/1

+2

✗

Correct answers

2 + (int*)

2+ (int*)

2 +(int*)

14 +

2 + (int*)

int *)((double*)



✖ line 18-Blank1

0/1

2+

✖

Correct answers

+8

+ 8

Problem 5

4 of 4 points

Choose if the following claims are true or false.

✓ 5.d. An algorithm that is implemented using a while loop can also be implemented using a do-while loop

1/1

☒ True

✓

☐ False

✓ 5.b. A recursive construct can have at most one Base Case(Stopping Condition).

1/1

☐ True

☒ False

✓



✓ 5.c.The recursive algorithm always consumes more memory than the iterative counterpart (if it exists). 1/1

☒ True



☐ False

Correct answer

☒ False

✓ 5.a. A recursive function must return a value for the recursion to work correctly. 1/1

☐ True

☒ False



Problem 6

3 of 3 points

✓ Blank2 1/1

1 2 0 1 2



✓ Blank3

1/1

2 0 1 2 0



Correct answers

2 0 1 2 0

2 0 1 2 0 #

✓ Blank1

1/1

0 1 2 0 1



Correct answers

0 1 2 0 1

0 1 2 0 1 #



Fill the output blanks of the following C program.

```
int a(int x);
int b(int y);
int c(int z);

int a(int x){
    if(x<1) return 0;
    return b(x-1);
}

int b(int y){
    if(y<1) return 1;
    return c(y-1);
}

int c(int z){
    if(z<1) return 2;
    return a(z-1);
}

int main(){
    for(int i=0;i<5;i+=1) printf("%d ", a(i)); printf("\n");
    for(int i=0;i<5;i+=1) printf("%d ", b(i)); printf("\n");
    for(int i=0;i<5;i+=1) printf("%d ", c(i));
}
```

Ans:

Problem 7

4 of 4 points

We are writing a code that, given a positive integer N, prints out all distinct partitions of N in increasing order. A partition of a positive integer n is a way of expressing n as a sum of positive integers.

Example N=4

OUTPUT: 1 1 1 1
1 1 2
1 3
2 2



✓ Line 11

1/1

```
return;
```



✗ Line 21

1/1

```
partition(n, 1, a, n, 0);
```



Correct answer

```
partition(n,1,a,n,0);
```

✗ Line 14

1/1

```
partition(n-i, i, a, max_value-i, count+1);
```



Correct answer

```
partition(n-i,i,a,max_value,count+1);
```

✓ Line 5

1/1

```
i<count
```



Fill in the blanks in the given code so that it outputs the partitions of input n.

```
1  #include <stdio.h>
2
3  void partition(int n, int min, int a[],int max_value, int count){
4      if(n==0){
5          for(int i=0;.....;i++)
6              printf("%d ",a[i]);
7          printf("\n");
8          return;
9      }
10
11      else if(n<0)
12          .....
13      for(int i=min;i<max_value;++i){
14          a[count] = i;
15          .....
16      }
17
18  int main(){
19      int n;
20      scanf("%d",&n);
21      int a[n];
22      .....
23      return 0;
24  }
```

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#)

Google Forms

