

```

1  /*
2   * Complete the 'myFunc' function below.
3   *
4   * The function is expected to return an
5   * The function accepts INTEGER n as para
6   */
7
8  int myFunc(int n)
9  {
10     while(n>1)
11     {
12         if(n%20==0)
13         {
14             n/=20;
15         }
16         else if(n%10==0)
17         {
18             n/=10;
19         }
20         else{
21             return 0;
22         }
23     }
24     return 1;
25 }
26
27

```

	Test	Expected	Got	
✓	printf("%d", myFunc(1))	1	1	✓
✓	printf("%d", myFunc(2))	0	0	✓
✓	printf("%d", myFunc(10))	1	1	✓
✓	printf("%d", myFunc(25))	0	0	✓
✓	printf("%d", myFunc(200))	1	1	✓

Passed all tests! ✓

```

1  /*
2   * Complete the 'powerSum' function below
3   *
4   * The function is expected to return an
5   * The function accepts following paramet
6   * 1. INTEGER x
7   * 2. INTEGER n
8   */
9
10 int powerSum(int x, int m, int n)
11 {
12     if(x==0)
13     {
14         return 1;
15     }
16     if(x<0)
17     {
18         return 0;
19     }
20     int count=0;
21     for(int i=m;;i++)
22     {
23         int power=1;
24         for(int j=0;j<n;j++)
25         {
26             power*=i;
27         }
28         if(power>x)
29         {
30             break;
31         }
32         count+=powerSum(x-power,i+1,n);
33     }
34     return count;
35 }

```

	Test	Expected	
✓	printf("%d", powerSum(10, 1, 2))	1	

Passed all tests! ✓