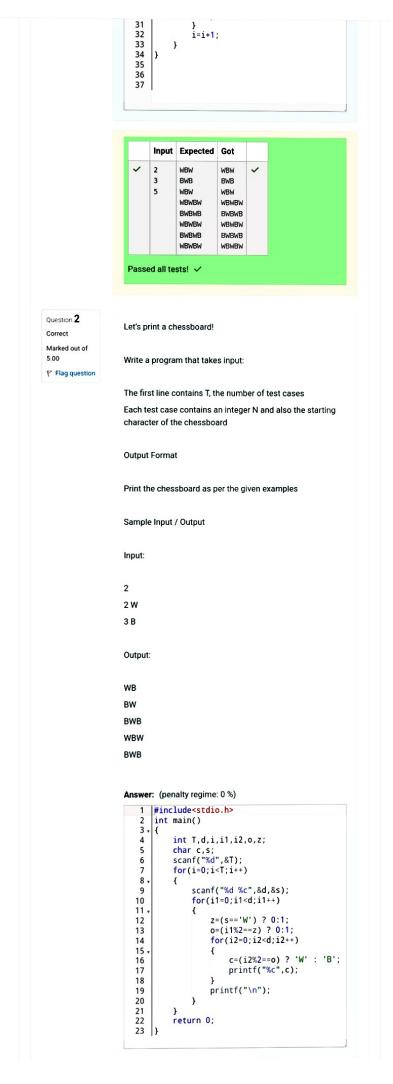
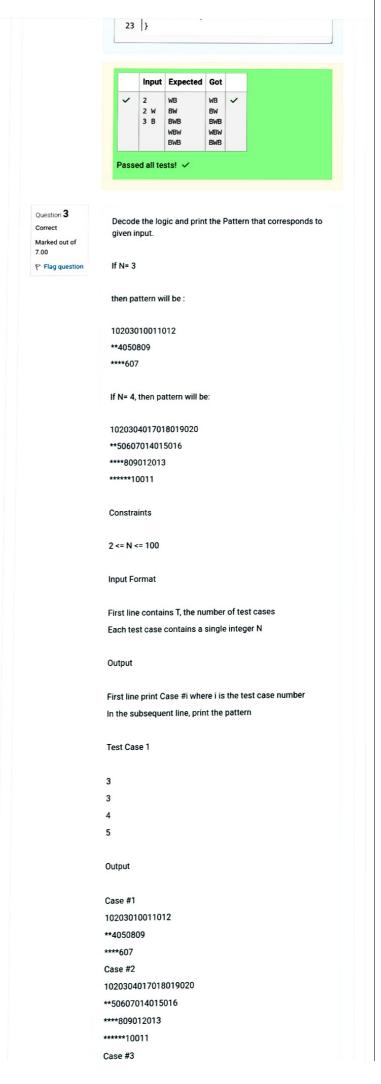
REC-CIS



```
Status Finished
           Started Monday, 23 December 2024, 5:33 PM
        Completed Saturday, 16 November 2024, 11:31 PM
          Duration 36 days 18 hours
Question 1
                   Write a program that prints a simple chessboard.
Correct
Marked out of
3.00
                   Input format:
F Flag question
                  The first line contains the number of inputs T.
                  The lines after that contain a different values for size of the
                  chessboard
                  Output format:
                  Print a chessboard of dimensions size * size. Print a Print W
                  for white spaces and B for black spaces.
                  Input:
                  2
                  3
                  5
                  Output:
                  WBW
                  BWB
                  WBW
                 WBWBW
                 BWBWB
                 WBWBW
                 BWBWB
                 WBWBW
                 Answer: (penalty regime: 0 %)
                     1 #include<stdio.h>
                         int main()
{
                     3 •
                     4
                              int T,d,i=0,i1,i2,o;
                             char c;
scanf("%d",&T);
                             while(i<T)
                     8
                     9
                                  scanf("%d",&d);
                    10
                                  i1=0;
                                  while(i1<d)
                    11
                    12
                    13
                                      0=1;
                    14
                                      i2=0;
                                      if(i1%2==0)
                    15
                    16
                                           o=0;
                    17
                    18
                                      while(i2<d)
                    19
                    20
                   21
22
                                           c='B';
                                           if(i2%2==o)
                   23 ·
24
                                                c='W';
                   25
                                           printf("%c",c);
                   26
                   27
                   28
                   29
                                      i1+=1;
                                      printf("\n");
                   30
                   31
                   32
                                 i=i+1;
                   33
                            }
                   34
                       }
                   35
                   36
```





```
3
 5
 Output
 Case #1
 10203010011012
 **4050809
 ****607
Case #2
 1020304017018019020
**50607014015016
****809012013
*****10011
Case #3
102030405026027028029030
**6070809022023024025
****10011012019020021
*****13014017018
*******15016
Answer: (penalty regime: 0 %)
     1 #include<stdio.h>
        int main() {
    int n,v,p3,c,in,i,i1,i2,t,ti;
    scanf("%d",&t);
    for(ti=0;ti<t;ti++){</pre>
    3
                   v=0;
                  scanf("%d",&n);
printf("Case #%d\n",ti+1);
for(i=0;i<n;i++){</pre>
    8
  10
                        if(i>0){
  11 -
                             for(i1=0;i1<i;i1++) print
  12
                 }
for(i1=i;i1<n;i1++){
  13
  14
                      if(i>0) c++;
printf("%d0",++v);
  15
  16
  17
                 if(i==0){
  18
                      p3=v+(v*(v-1))+1;
  19
  20
                      in=p3:
  21
                in=in-c;
  23
 24
25
26
27
                ps-II,
for(i2=i;i2<n;i2++){
    printf("%d",p3++);
    if(i2!=n-1) printf("0");
}printf("\n");</pre>
  28
  29
  30
  31
```

```
Input Expected
                                      Got
      3
            Case #1
                                      Case #1
            10203010011012
                                      102030100110
      3
            **4050809
                                      **4050809
            ****607
                                      ****607
            Case #2
                                      Case #2
            1020304017018019020
                                      102030401701
            **50607014015016
                                      **5060701401
            ****809012013
                                      ****80901201
            *****10011
                                      *****10011
            Case #3
                                      Case #3
            102030405026027028029030
                                     102030405026
            **6070809022023024025
                                      **6070809022
            ****10011012019020021
                                      ****10011012
            *****13014017018
                                      *****130140
            ******15016
                                      ******1501
Passed all tests! ✓
```

Quiz navigation

1 2 3

Show one page at a time

Finish review

Status Finished Started Monday, 23 December 2024, 5:33 PM Completed Sunday, 17 November 2024, 12:07 AM Duration 36 days 17 hours Question 1 The k-digit number N is an Armstrong number if and only if Correct the k-th power of each digit sums to N. Marked out of 3.00 Given a positive integer N, return true if and only if it is an P Flag question Armstrong number. Example 1: Input: 153 Output: true Explanation: 153 is a 3-digit number, and 153 = 1^3 + 5^3 + 3^3. Example 2: Input: 123 Output: false Explanation: 123 is a 3-digit number, and 123 != 1^3 + 2^3 + 3^3 = 36. Example 3: Input: 1634 Output: true Note: 1 <= N <= 10^8 Answer: (penalty regime: 0 %) 1 Wineludocetdia ha

```
1 <= N <= 10^8
                    Answer: (penalty regime: 0 %)
                            #include<stdio.h>
                            #include<math.h>
                        3
                            int main()
                        4 ,
                                 int n;
scanf("%d",&n);
                                 int x=0, n2=n;
                        8
                                 while(n2!=0)
                       10
                                     n2=n2/10;
                       12
                       13
                                 int sum=0;
                       14
15
                                 int n3=n,n4;
                                while(n3!=0)
                       16
                       17
                                     n4=n3%10;
                                     sum = sum+pow(n4,x);
n3=n3/10;
                       19
                       20
                                if(n==sum)
{
                       21
22
                       23
                                     printf("true");
                       24
                       25
                                 else
                      26
27
                                     printf("false");
                       28
                       29
                                return 0;
                       30
                       31 }
                           Input Expected Got
                           153
                                  true
                                             true
                           123
                                  false
                                            false
                     Passed all tests! ~
Question {f 2}
                   Take a number, reverse it and add it to the original number
Correct
                   until the obtained number is a palindrome. Constraints
Marked out of
                   1<=num<=99999999 Sample Input 1 32 Sample Output 1 55
5.00
                   Sample Input 2 789 Sample Output 2 66066
F Flag question
                   Answer: (penalty regime: 0 %)
                          #include<stdio.h>
                       2
                           int main()
                       3 ,
                       4
                                int rn,n,nt=0,i=0;
scanf("%d",&n);
                       5
6
7
                                    nt=n;rn=0;
                                    while(n!=0)
                       8
                                         rn=rn*10 + n%10;
                      10
                      11
                                         n=n/10;
                      12
                      13
                                     n=nt+rn;
                      14
                                    i++;
                      15
                               }
while(rn!=nt || i==1);
printf("%d",rn);
                      16
                      17
                      18
19
                                return 0;
                          Input Expected Got
                                  55
                          789
                                 66066
                                            66066
                    Passed all tests! 🗸
```

Question 3
Correct
Marked out of 7.00
P Flag question

A number is considered lucky if it contains either 3 or 4 or 3 and 4 both in it. Write a program to print the nth lucky number. Example, 1st lucky number is 3, and 2nd lucky number is 4 and 3rd lucky number is 33 and 4th lucky number is 34 and so on. Note that 13, 40 etc., are not lucky as they have other numbers in it.

The program should accept a number 'n' as input and display the nth lucky number as output.

Sample Input 1:

3

Sample Output 1:

33

Explanation:

Here the lucky numbers are 3, 4, 33, 34., and the 3rd lucky number is 33.

Sample Input 2:

34

Sample Output 2:

33344

```
Answer: (penalty regime: 0 %)
```

```
#include<stdio.h>
 2
     int main()
         int n=1,i=0,nt,co=0,e;
scanf("%d",&e);
 4
 5
         while(i<e)
              nt=n;
 8
              while(nt!=0)
9
10
11
                  co=0;
                  if(nt%10!=3 && nt%10!=4)
12
13
14
                       co=1;
                      break;
15
16
                  nt=nt/10;
17
18
19
              if(co==0)
20
21
22
23
              n++;
24
         printf("%d",--n);
25
         return 0;
26
27 }
```

