Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,r=0;
    scanf("%d",&n);
    while(n!=0)
    {
        n=n/2;
        r=r+1;
    }
    printf("%d",r);
}
```

	Input	Expected	Got	
✓	10	4	4	✓
✓	5	3	3	✓
✓	20	5	5	✓
✓	500	9	9	✓
✓	1000	10	10	✓

Passed all tests! ✓

6

7

Output

Yes

Yes

No

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int T,i=0,n,t;
    scanf("%d",&T);
    while(i<T)
    {
        scanf("%d",&n);
        t=n/4;
        if(t%2==0&&n%2==0)
        {
            printf("No\n");
        }
        else if(t%2==1&&n%2==1)
        {
            printf("No\n");
        }
        else
        {
            printf("Yes\n");
        }
        i++;
    }
}
```

	Input	Expected	Got	
✓	3 1 6 7	Yes Yes No	Yes Yes No	✓

Passed all tests! ✓

Sample Input

1288

Sample Output

4

Explanation

Add the holes count for each digit, 1, 2, 8, 8. Return 0 + 0 + 2 + 2 = 4.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int a,i,n=0;
    scanf("%d",&a);
    while(a>0)
    {
        i=a%10;
        if(i==0||i==6||i==4||i==9)
        {
            n=n+1;
        }
        else if(i==8)
        {
            n=n+2;
        }
        a=a/10;
    }
    printf("%d",n);
}
```

	Input	Expected	Got	
✓	630	2	2	✓
✓	1288	4	4	✓

Sample Output 2

5

Explanation 2

2 + 3 = 5, is the best case for maximum nutrients.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    long long int n,t,i,nut=0;
    scanf("%lld %lld",&n,&t);
    for(i=1;i<=n;i++)
    {
        nut =nut+i;
        if(nut==t)
        {
            nut=nut-1;
        }
    }
    printf("%lld",nut%1000000007);
}
```

	Input	Expected	Got	
✓	2 2	3	3	✓
✓	2 1	2	2	✓
✓	3 3	5	5	✓

Passed all tests! ✓

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,x,y=1;
    scanf("%d",&n);
    while(n!=0&&y==1)
    {
        x=n%10;
        n=n/10;
        if(x==2||x==3||x==4||x==7)
        {
            y++;
        }
    }
    if(y==1)
    {
        printf("true");
    }
    else
    {
        printf("false");
    }
}
```

	Input	Expected	Got	
✓	6	true	true	✓
✓	89	true	true	✓
✓	25	false	false	✓

Passed all tests! ✓

Input:

5 10 15 20 25 30 35 40 45 50

Output:

5

Explanation:

The numbers meeting the criteria are 5, 15, 25, 35, 45.

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int n,x=0;
    while(scanf("%d",&n)==1)
    {
        if(n%2!=0)
        {
            x++;
        }
    }
    printf("%d",x);
}
```

	Input	Expected	Got	
✓	5 10 15 20 25 30 35 40 45 50	5	5	✓

Passed all tests! ✓