Lesson 76 Practice 19

You need to develop the ability of creativity and thinking and work on it to develop it, because it will be a very important factor for the programmer.

For example: in the labor market, if a client asked the programmer for a specific program and he did it with the same specifications, then this programmer is not a creative person, but if he makes some additions in the program which benefits the customer, the price of this program will increase, anything that has an advantage that is not found in the rest of the programs or anything else has price higher than the rest.

We need to write a program, when the user enters numbers but when he enters -1 the program stops.

In this question he did not mention to us what we will do?

Are we going to count the positive numbers, negative numbers, even or odd numbers

Shall we calculate -1 with numbers

But here to develop creativity and thinking ability, we can add a lot of these additions. We can solve the question in this way:

```
#include <stdio.h>

int main(){

int x, n = 0;

while (scanf("%d", &x) && x != -1)

//because we don't know how many number
```

```
//he will enter before entering -1
n++;
printf("%d", n);
// no need for an array
}
```

input:

3

6

5

2

1

4

-1

output:

6

Try the code yourself: Click Here!

Here we didn't add the number -1, let's try counting now :

```
#include <stdio.h>
int main(){
int x, n = 1;
while (scanf("%d", &x) && x != -1)
n++;
```

```
printf("%d", n);
}
```

input:

3

6

5

2

1

4

-1

output:

7

Try the code yourself: Click Here!

If we want to calculate the **even** numbers, it will be as follows:

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

```
printf("%d", n);
}
```

input:

3

6

5

2

4

1

9

8

7

-1

output:

4

Try the code yourself: Click Here!

If we want to calculate the **odd** numbers, it will be as follows:

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

```
n++;
printf("%d", n);
}
```

input:

```
3
6
5
2
4
1
9
8
7
-1
```

output:

```
5
```

Try the code yourself: Click Here!

If we want to know if the number is **palindrome** :

```
#include <stdio.h>
int ispal(int x) {
int y = x, z=0;
while (y) {
```

```
z = z * 10 + y % 10;
y /= 10;
}
return x == z;
//if true it will return 1
}
int main() {
int x, n = 0;
while (scanf("%d", &x) && x != -1)
if (ispal(x))
n++;
printf("%d", n);
}
```

input:

121

236

598

565

-1

output:

2

Try the code yourself: [Click Here!]