

Leer caracter por teclado desues de leer un entero

Escrito por miloarias - 12/05/2015 21:32

Hola a todos,

Alguien sabe porque despues de haber leido un dato entero en el siguiente codigo cuando intento leer un caracter simple al parecer no se ejecuta la linea del scanf();.

```
#include
#include

int main()
{
    int salario;
    char letra;

    printf("Ingrese el Sueldo Bruto del Empleado: ");
    scanf("%i", &salario);

    printf("Se Encuentra Afiliado a Salud s/n: ");
    scanf("%c", &letra);

    return 0;
}
```

Si coloco las dos instrucciones en donde se lee el caracter antes de la donde se lee el entero, el codigo se ejecuta como se espera, pero colocando las instrucciones tal y como estan en el codigo que copie, no me da para escribir el caracter y la ejecuci3n finaliza.

Espero alguien pueda resolver mi duda.

Saludos.

=====

Re: Leer caracter por teclado desues de leer un entero

Escrito por ajo - 13/05/2015 17:48

Lo que necesitas es agregar la funci3n fflush, te pongo un ejemplo:

```
printf("Ingrese el Sueldo Bruto del Empleado: ");
scanf("%i", &salario);

/*limpiar el buffer*/
fflush(stdin);

printf("Se Encuentra Afiliado a Salud s/n: ");
scanf("%c", &letra);
```

```
printf("Valor leído: %dn",salario);  
printf("Caracter leído: %cn",letra);
```

=====

Re: Leer caracter por teclado desues de leer un entero

Escrito por miloarias - 15/05/2015 14:58

Si, con esa función si me ejecuto con normalidad el codigo.

¿Pero porque limpiar el buffer? No se supone que las dos variables pertenecen a posiciones de memoria diferentes y en cada una de estas posiciones se debe de almacenar el respectivo tipo de dato.
¿Eso es mas cuestión en este caso del lenguaje de programación o que?.

Saludos.

=====

Re: Leer caracter por teclado desues de leer un entero

Escrito por adrianvaca - 01/06/2015 07:28

Normalmente lo que lees de teclado se coloca en un bufer que es una especie de memoria intermedia.

Aunque a veces no hace falta, se suele limpiar el bufer para asegurarte que lo que lees de teclado corresponde al último ingreso y no a uno anterior.

Para limpiar el bufer se suele colocar lo siguiente antes de invocar a scanf, gets entre otras:

```
fflush(stdin);
```

=====