



INFORMATICA I

Uso de la función SCANF y PRINTF

Ing. Juan Carlos Cuttitta

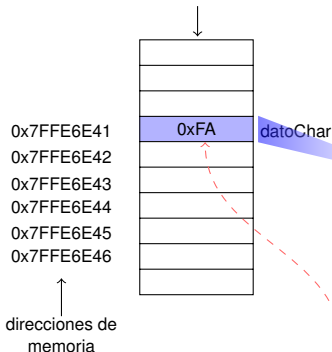
*Universidad Tecnológica Nacional
Facultad Regional Buenos Aires
Departamento de Ingeniería Electrónica*

10 de abril de 2020

Función scanf para tipo de dato *char*

Arquitectura X86 32 bits

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5 char datoChar;
6 .....
7 .....
8 return (0);
9 }
```

La memoria inicialmente puede
contener cualquier valor !!!

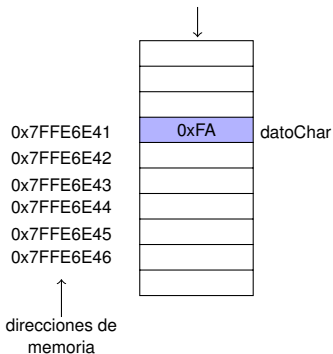
Reserva un espacio de
memoria tipo *char*



UTN.BA

Función scanf para tipo de dato *char*

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     .....
10    .....
11    return (0);
12 }
```

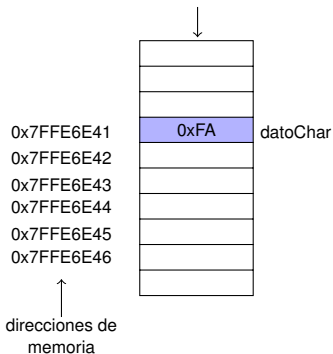
Imprime en pantalla una
leyenda para el usuario



UTN.BA

Función scanf para tipo de dato *char*

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5 char datoChar;
6
7 //— Ingreso de datos —
8 printf("Ingreso valor : ");
9 scanf("%c", &datoChar);
10 .....
11 .....
12 return (0);
13 }
```

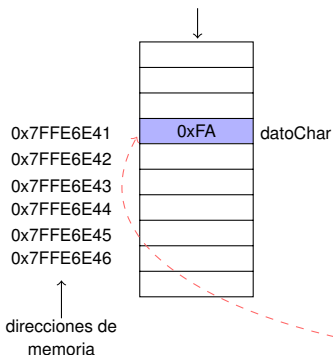
Indica el **tipo de dato** que va a guardar en memoria



UTA.BA

Función scanf para tipo de dato *char*

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5 char datoChar;
6
7 //— Ingreso de datos —
8 printf("Ingreso valor : ");
9 scanf("%c", &datoChar);
10 .....
11 .....
12 return (0);
13 }
```

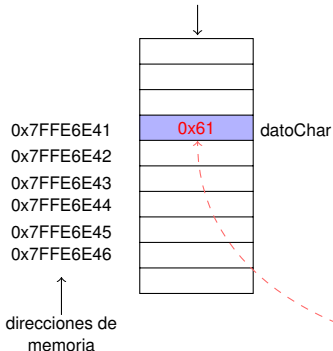
Indica la dirección de memoria donde va a guardar el dato



UTA.BA

Función scanf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     scanf("%c", &datoChar);
10    .....
11    .....
12    return (0);
13 }
```

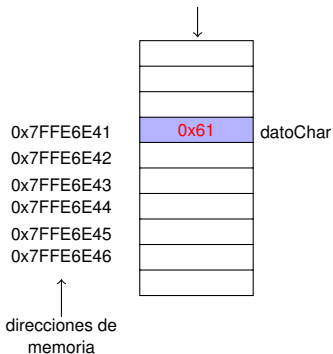
El dato ingresado fué
el caracter 'a'



UTN.BA

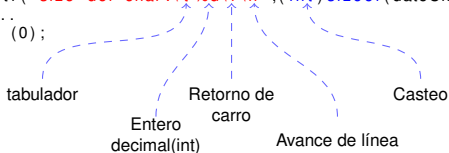
Función printf para tipo de dato *char*

Disposición de la variable
datoChar en memoria



Código en programa fuente

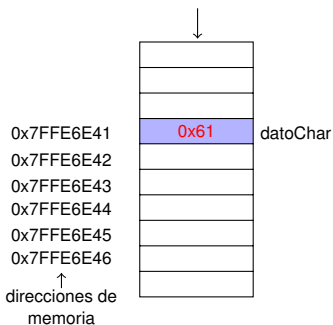
```
1 #include <stdio.h>
2
3 int main (void)
4 {
5 char datoChar;
6
7 //— Ingreso de datos —
8 printf("Ingreso valor : ");
9 scanf("%c", &datoChar);
10 //— Impresión de datos —
11 printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
12 .....
13 return (0);
14 }
```



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     /*— Ingreso de datos —
8     printf("Ingrese valor : ");
9     scanf("%c", &datoChar);
10    /*— Impresión de datos —
11    printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
12    .....
13    return (0);
14 }
```

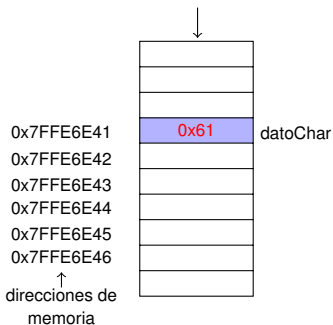
Ingrese valor : a
size del char : 1



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     //— Ingreso de datos —
8     printf("Ingrese valor : ");
9     scanf("%c", &datoChar);
10    //— Impresión de datos —
11    printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
12    .....
13    return (0);
14 }
```

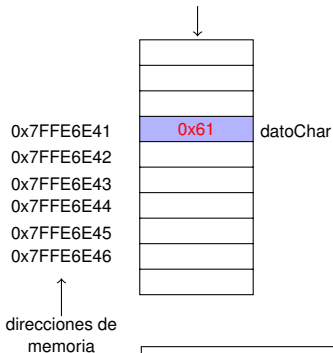
Ingrese valor : a
size del char : 1



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     scanf("%c", &datoChar);
10    //— Impresión de datos —
11    printf("size del char:\t%d\r\n", (int)sizeof(datoChar));
12    printf("el dato en decimal:\t%d\r\n", datoChar);
13    .....
14    return (0);
15 }
```

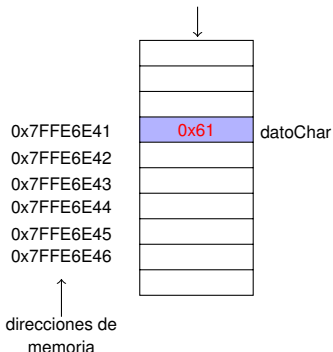
```
Ingreso valor : a
size del char      : 1
el dato en decimal : 97
```



UTN.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     /*— Ingreso de datos —
8     printf("Ingrese valor : ");
9     scanf("%c", &datoChar);
10    /*— Impresión de datos —
11    printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
12    printf("el dato en decimal:\t%d\r\n", datoChar);
13    printf("el dato en hexa:\t%x\r\n", datoChar);
14    .....
15    return (0);
16 }
```

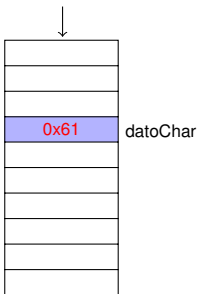
```
Ingrese valor : a
size del char      : 1
el dato en decimal : 97
el dato en hexa   : 61
```



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     scanf("%c", &datoChar);
10    //— Impresión de datos —
11    printf("size del char:\t%d\r\n", (int)sizeof(datoChar));
12    printf("el dato en decimal:\t%d\r\n", datoChar);
13    printf("el dato en hexa:\t %#x\r\n", datoChar);
14    .....
15    return (0);
16 }
```

↑
direcciones de
memoria

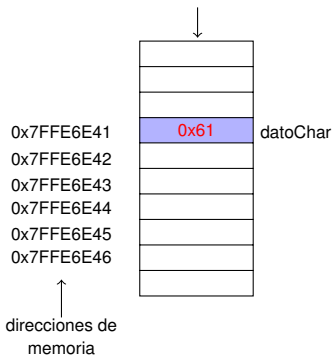
Ingreso valor : a
size del char : 1
el dato en decimal : 97
el dato en hexa : 0x61



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5 char datoChar;
6 //— Ingreso de datos —
7 printf("Ingreso valor : ");
8 scanf("%c", &datoChar);
9 //— Impresión de datos —
10 printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
11 printf("el dato en decimal:\t%d\r\n", datoChar);
12 printf("el dato en hexa:\t%x\r\n", datoChar);
13 printf("el dato en octal:\t%o\r\n", datoChar);
14 return (0);
15 }
```

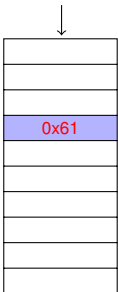
Ingreso valor : a	
size del char	: 1
el dato en decimal	: 97
el dato en hexa	: 0x61
el dato en octal	: 141



UTA.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



0x7FFE6E41
0x7FFE6E42
0x7FFE6E43
0x7FFE6E44
0x7FFE6E45
0x7FFE6E46

↑
direcciones de memoria

datoChar

Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6     //— Ingreso de datos —
7     printf("Ingreso valor : ");
8     scanf(" %c", &datoChar);
9     //— Impresión de datos —
10    printf("size del char:\t%d\r\n", (int) sizeof(datoChar));
11    printf("el dato en decimal:\t%d\r\n", datoChar);
12    printf("el dato en hexa:\t %#x\r\n", datoChar);
13    printf("el dato en octal:\t %o\r\n", datoChar);
14    printf("dirección en memoria:\t %p\r\n", &datoChar);
15    return (0);
16 }
```

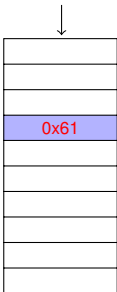
Ingreso valor : a	:	
size del char	:	1
el dato en decimal	:	97
el dato en hexa	:	0x61
el dato en octal	:	141
dirección en memoria:	:	0x7FFE6E41



UTN.BA

Función printf para tipo de dato **char**

Disposición de la variable
datoChar en memoria



0x7FFE6E41
0x7FFE6E42
0x7FFE6E43
0x7FFE6E44
0x7FFE6E45
0x7FFE6E46

datoChar

↑
direcciones de
memoria

Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     char datoChar;
6     //— Ingreso de datos —
7     printf("Ingrese valor : ");
8     scanf("%c", &datoChar);
9     //— Impresión de datos —
10    printf("size del char:\t%d\r\n", (int)sizeof(datoChar));
11    printf("el dato en decimal:\t%d\r\n", datoChar);
12    printf("el dato en hexa:\t %#x\r\n", datoChar);
13    printf("el dato en octal:\t %o\r\n", datoChar);
14    printf("dirección en memoria:\t%p\r\n", &datoChar);
15    printf("el caracter es:\t%c\r\n", datoChar);
16    return (0);
17 }
```

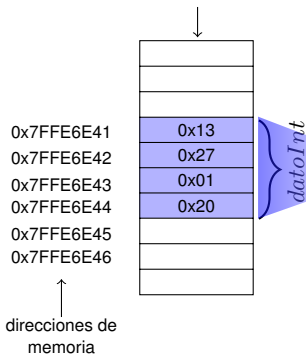
```
Ingrese valor : a
size del char      : 1
el dato en decimal : 97
el dato en hexa   : 0x61
el dato en octal  : 141
dirección en memoria: 0x7FFE6E41
el caracter es    : a
```



UTA.BA

Función scanf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6     .....
7     .....
8     return (0);
9 }
```

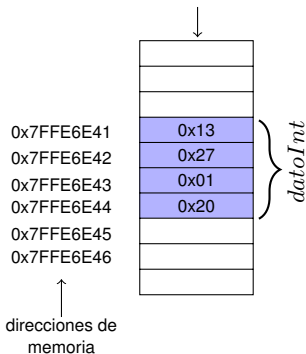
*Reserva un espacio de memoria tipo **int***



UTA.BA

Función scanf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     //— Ingreso de datos —
8     printf("Ingrese valor : ");
9     .....
10    .....
11    return (0);
12 }
```

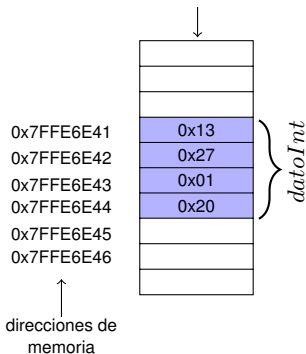
Imprime en pantalla una
leyenda para el usuario



UTN.BA

Función scanf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     scanf("%d", &datoInt);
10    .....
11    .....
12    return (0);
13 }
```

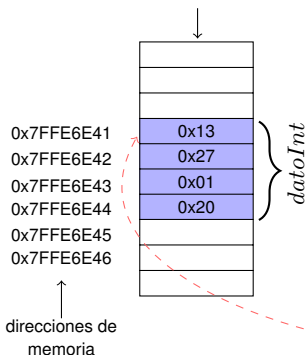
Indica el **tipo de dato** que
toma del teclado



UTA.BA

Función scanf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     //— Ingreso de datos —
8     printf("Ingreso valor : ");
9     scanf("%d", &datoInt);
10    .....
11    .....
12    return (0);
13 }
```

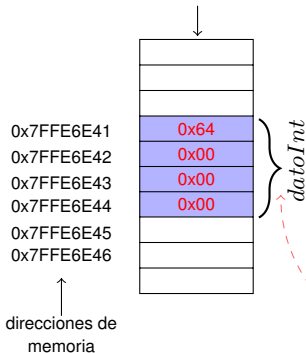
Indica la dirección de memoria
donde va a guardar el dato



UTA.BA

Función scanf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     //— Ingreso de datos —
8     printf("Ingrese valor : ");
9     scanf("%d", &datoInt);
10    .....
11    .....
12    return (0);
13 }
```

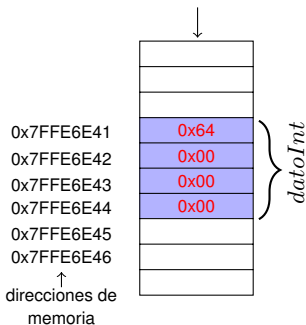
El dato ingresado fué
el número 100



UTA.BA

Función printf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     /*— Ingreso de datos —
8     printf("Ingrese valor : ");
9     scanf("%d", &datoInt);
10    /*— Impresión de datos —
11    printf("size del int:\t%d\r\n", (int)sizeof(datoInt));
12    .....
13    return (0);
14 }
```

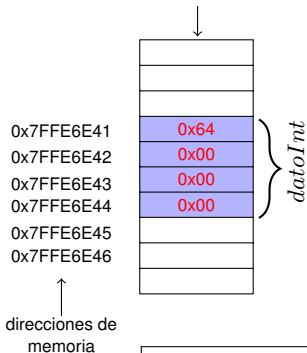
```
Ingrese valor : 100
size del int      : 4
```



UTA.BA

Función printf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     /* Ingreso de datos */
8     printf("Ingreso valor : ");
9     scanf("%d", &datoInt);
10    /* Impresión de datos */
11    printf("size del int:\t%d\r\n", (int)sizeof(datoInt));
12    printf("el dato en decimal:\t%d\r\n", datoInt);
13    .....
14    return (0);
15 }
```

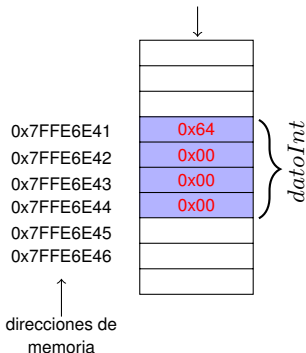
```
Ingreso valor : 100
size del int      : 4
el dato en decimal : 100
```



UTN.BA

Función printf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



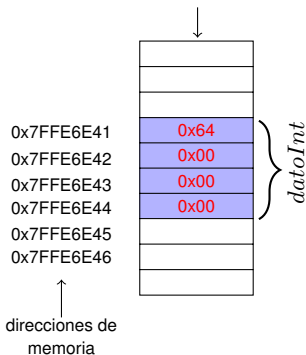
Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6
7     /* Ingreso de datos */
8     printf("Ingrese valor : ");
9     scanf("%d", &datoInt);
10    /* Impresión de datos */
11    printf("size del int:\t%d\r\n", (int)sizeof(datoInt));
12    printf("el dato en decimal:\t%d\r\n", datoInt);
13    printf("el dato en hexa:\t %#x\r\n", datoInt);
14    .....
15    return (0);
16 }
```

```
Ingrese valor : 100
size del int      : 4
el dato en decimal : 100
el dato en hexa   : 0x64
```

Función printf para tipo de dato *int*

Disposición de la variable
datoInt en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     int datoInt;
6     //— Ingreso de datos —
7     printf("Ingrese valor : ");
8     scanf("%d", &datoInt);
9     //— Impresión de datos —
10    printf("size del int:\t%d\r\n", (int)sizeof(datoInt));
11    printf("el dato en decimal:\t%d\r\n", datoInt);
12    printf("el dato en hexa:\t%x\r\n", datoInt);
13    printf("dirección en memoria:\t%p\r\n", &datoInt);
14    return (0);
15 }
```

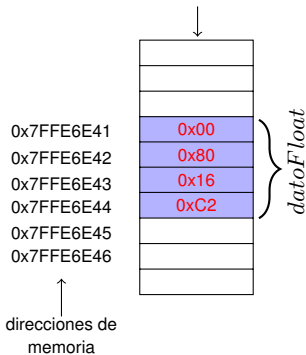
```
Ingrese valor : 100
size del int      : 4
el dato en decimal : 100
el dato en hexa   : 0x64
dirección en memoria: 0x7FFE6E41
```



UTA.BA

Función printf para tipo de dato **float**

Disposición de la variable
datoFloat en memoria



Código en programa fuente

```
1 #include <stdio.h>
2
3 int main (void)
4 {
5     float datoFloat;
6     //— Ingreso de datos —
7     printf("Ingreso valor : ");
8     scanf("%f", &datoFloat);
9     //— Impresión de datos —
10    printf("size del float : \t%d\r\n", (int) sizeof(datoFloat));
11    printf("el dato en real : \t%f\r\n", datoFloat);
12    printf("el dato en real : \t%0.2f\r\n", datoFloat);
13    return (0);
14 }
```

```
Ingreso valor : 3.1415
size del float   :    4
el dato en real  :   3.1415
el dato en real  :   3.14
```



UTN.BA