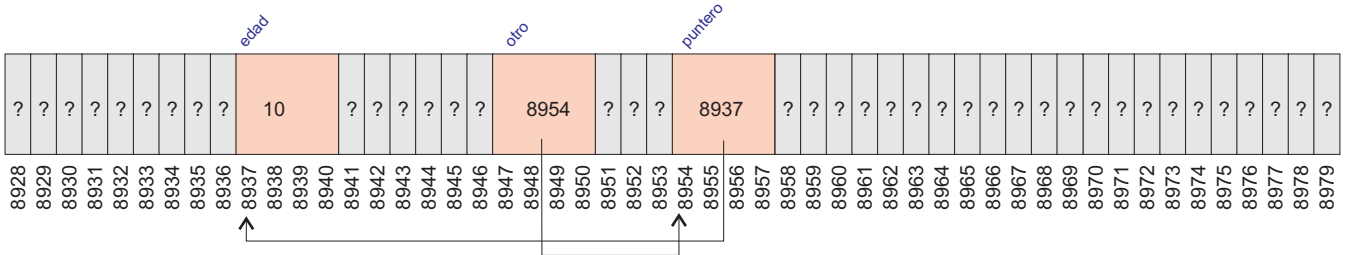


```
int * puntero = &edad;
// se asigna memoria para almacenar un puntero a un
// entero (también 4 bytes), y se inicializa ese puntero con
// la dirección de memoria donde se encuentra edad (8937)
```

```
// la variable puntero (8934)
```



```
printf("%d", otro);  
// Se imprime el valor de la variable otro : 8954
```

```
printf("%d", &otro),
// Imprime la dirección de memoria donde se encuentra la variable otro
// 8947
```

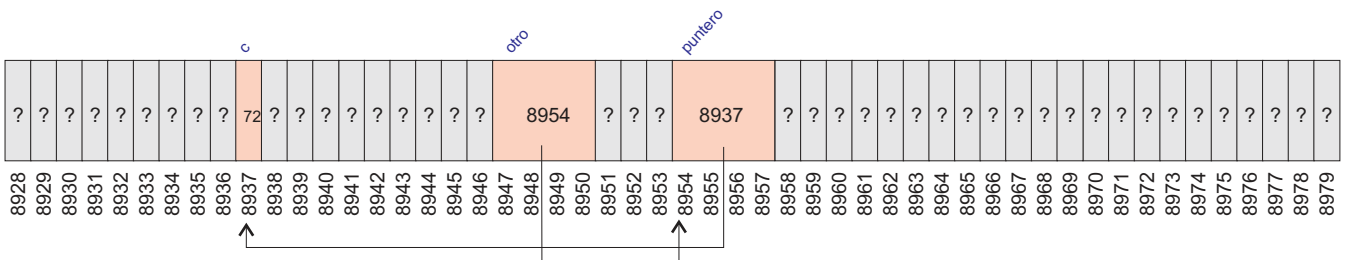
```
printf("%d", *otro);  
    // Imprime el valor de aquello a lo que apunta otro, que es la variable  
    // puntero : 8937
```

```
printf("%d", **otro);
// Mira a qué apunta otro (8954), y ese valor (8937) a su vez lo toma como un
// puntero, imprimiendo lo que hay en esa posición de memoria :10
```

```
// se asigna memoria para almacenar un char (1 bytes) ,y
// se inicializa a 72 (el código ASCII de 'H')
```

```
char* puntero = &edad;  
    // se asigna memoria para almacenar un puntero a un  
    // char (4 bytes), y se inicializa ese puntero con  
    // la dirección de memoria donde se encuentra c (8937)
```

```
// se asigna memoria para almacenar un puntero a un puntero a un
// char, y se le asigna la dirección de memoria donde se encuentra
// la variable puntero (8954)
```



```
printf("%d", otro+1);  
    // otro apunta a la variable puntero. otro+1 por lo tanto es la  
    // dirección de memoria donde se guardaría la siguiente variable "puntero"  
    // Como la variable puntero ocupa 4 bytes, otro+1 = 8954+4 = 8958
```

```
printf("%d", *otro+1),
// *otro es el valor de aquello a lo que apunta otro, que es por lo tanto
// también el valor de puntero (8937). puntero apunta a c con lo que
// *otro+1 = puntero + 1 = el sitio donde se guardaría la siguiente variable c.
// Como la variable c ocupa un byte, *otro+1 = puntero + 1 = 8937 + 1 = 8938
```

```
printf("%d", **otro+1);
// **otro es el valor de la variable c. Como c ya no es un puntero
// de ningún tipo, **otro + 1 simplemente imprime el valor de esa
// variable aumentado en 1 : 73
```