

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5     int var_a;           // Variable entera
6     int *p_var_a;        // Puntero a entero
7
8     puts("=====");
9     puts("          PUNTEROS          ");
10    puts("=====");
11
12    /* 1. Leer un valor para la variable var_a */
13    printf("\nIntroduce un valor: ");
14    scanf("%d", &var_a);
15
16    /* 2. Escribir contenido de la variable var_a */
17    printf("\nEl contenido de var_a es: %d", var_a);
18
19    /* 3. Apuntar a variable var_a con el puntero p_var_a */
20    p_var_a = &var_a;
21
22    /* 4. Utilizar puntero p_var_a para escribir el contenido de la variable
23    var_a */
24    printf("\nEl contenido de la variable var_a (desde p_var_a) es: %d",
25    *p_var_a);
26
27    /* 5. Utilizar el puntero p_var_a para leer un valor para la variable
28    var_a */
29    printf("\n\nValor para var_a (desde p_var_a): ");
30    scanf("%d", p_var_a);
31
32    /* 6. Escribir el contenido de la variable var_a */
33    printf("\nEl contenido de var_a es: %d", var_a);
34
35    /* 7. Utilizar el puntero p_var_a para escribir el contenido de la
36    variable var_a */
37    printf("\nEl contenido de var_a (desde p_var_a) es: %d", *p_var_a);
38
39    /* 8. Escribir la dirección de la variable var_a utilizando el operador
40    dirección */
41    printf("\nLa direccion de var_a es: %p", &var_a);
42
43    /* 9. Utilizar el puntero p_var_a para escribir la dirección de la
44    variable var_a */
45    printf("\nLa direccion de var_a (desde p_var_a) es: %p", p_var_a);
46
47    /* 10. Escribir la dirección de la variable p_var_a */
48    printf("\nLa dirección de p_var_a es: %p", &p_var_a);
49
50    /* 11. Escribir el contenido de la variable p_var_a */
51    printf("\nEl contenido de p_var_a es: %d", *p_var_a);
52
53    puts("");
54    system("pause");
55    return 0;
56 }
```