

UT 5: Estructuras de datos dinámicas

Tarea 1

1. ¿Cuál es la salida de los siguientes programas? Primero piensa cuál debe ser la salida y luego copia el programa en Visual Studio Code para comprobarlo.

```
#include <stdio.h>

int main()
{
    char c1 = 'A';
    char c2 = 'B';
    char *pc;
    pc = &c1;
    printf("%c\n", c1); // se imprime: ?
    *pc = 'a';
    printf("%c\n", c1); // se imprime: ?
    c1 = 'Z';
    printf("%c\n", *pc); // se imprime: ?
    pc = &c2;
    printf("%c\n", *pc); // se imprime: ?
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    int f;
    int *pe;
    pe = &f;
    for (*pe = 1; *pe <= 10; *pe = *pe + 1)
    {
        printf("%i\n", f); // se imprime ?
    }
    return 0;
}
```

```
#include <stdio.h>

int main()
{
    float z1, z2;
    float *pf;
    pf = &z1;
    *pf = 10.2;
    pf = &z2;
    *pf = 20.90;
    printf("%.2f %.2f", z1, z2); // se imprime ?
    return 0;
}
```

```

#include <stdio.h>

int main()
{
    int vec[5] = {10, 20, 30, 40, 50};
    int *pe;
    pe = vec;
    printf("%i\n", *pe); // ?
    pe++;
    printf("%i\n", *pe); // ?
    pe++;
    printf("%i\n", *pe); // ?
    pe--;
    printf("%i\n", *pe); // ?
    return 0;
}

```

2. Se define un array de dos enteros de 4 bytes del tipo `int array[2]` y está almacenado a partir de la posición 100 de memoria. Se ejecuta el código:
- ```
array[0]=20;
array[1] = 30;
```

- ¿Qué valor tiene el entero almacenado en la posición de memoria 104?
- ¿Y en qué dirección está almacenado el primer elemento del array?

3. En un programa se definen estas dos variables:

```
int i = 10;
int *i_ptr;
```

- a. Si la variable `i` está almacenada en la posición de memoria 100, ¿qué valor contiene la variable `i_ptr`?
4. Se define las variables `a` de tipo entero, `b` de tipo puntero a entero, y `c` de tipo puntero a puntero a entero. ¿Cómo se consigue que `c` tenga la dirección de la dirección del entero `a`?
- `c = b; b = a;`
  - `c = b; b = &a;`
  - `c = &b; b = a;`
  - `c = &b; b = &a;`

5. Dadas dos variables de cualquier tipo en un programa en C, considera la siguiente expresión:

```
&a == &b;
```

- La expresión es incorrecta.
- La expresión es correcta y es siempre falsa.

- c. La expresión es correcta y es siempre cierta.
  - d. La expresión es correcta y es cierta o false dependiendo de los valores de a y b.
6. Dado el siguiente código, ¿que valor devolvería las variables y, z, nptr y mptr si las imprimieras antes del return 0? Copia el programa y modifícalo para añadir los printf de las variables y comprueba si la solución obtenida es la esperada.

```
#include <stdio.h>

int main()
{
 int y = 5;
 int z = 3;
 int *nptr;
 int *mptr;
 nptr = &y;
 z = *nptr;
 *nptr = 7;
 mptr = nptr;
 mptr = &z;
 *mptr = *nptr;
 y = (*nptr) + 1;
 return 0;
}
```