

# TEMA 6: TIPOS DE DATOS ESTRUCTURADOS

---

---

## CUESTIONES

---

---

1. Sea la siguiente declaración de tipos

```
typedef float Vector[5];
struct registro
{
    string nombre;
    Vector datos;
}
typedef registro Vec[10][20];
```

Si X es una variable de tipo Vec, señalar los accesos correctos y el tipo de la componente accedida:

Acceso	Tipo
X[1]	-----
X[1][15].nombre	-----
X[2][1][1].datos	-----
X[1][1].datos[3]	-----
X.datos[5]	-----
X[0][1].nombre[1]	-----

2. Dado el siguiente programa:

```
string agrega(string & s)
{
    string p;

    p = "como estamos";
    s = s + p;
    return (s + "!");
}

int main()
{
    string s;

    s = "Hola ";
    s = agrega(s);
    cout << s;
    return 0;
}
```

Indica la salida correcta:

- a) 'Hola como estamos!'
- b) 'Hola Hola como estamos!'
- c) 'como estamos Hola!'
- d) Se produce un error de ejecución.

3. La declaración de variable:

```
typedef elemento tabla[t1][t2];
tabla t;
```

- a) equivale a:  
elemento t[t1][t2];
- b) equivale a:  
typedef elemento vector[t2];  
typedef vector tabla[t1];  
tabla t;
- c) a y b son ciertas.
- d) a y b son falsas.

4. Dado el siguiente programa:

```
typedef int Vector[10];

int main()
{
    Vector v;
    int i;

    v[0] = 1;
    for(i = 0; i < 10; i++)
        v[i + 1] = v[i] + 1;
    cout << v[1] << v[5];
    return 0;
}
```

Indicar qué salida visualiza en caso de ser correcto, o qué tipo de error existe en caso contrario.

5. Dadas las siguientes declaraciones:

```
typedef int tv1[10];
typedef int tv2[10];
typedef int tv3[10][2];

tv1 v1;
tv2 v2, v4;
tv3 v3;
int v5[10];
tv1 v6;
```

a) Indicar que variables son equivalentes a v1 utilizando un criterio de equivalencia de tipos por nombre.

b) Indicar que variables son equivalentes a v1 utilizando un criterio de equivalencia de tipos estructural.

6. Dada la siguiente matriz:

```
float matriz[5][5];
```

Y sabiendo que la variable matriz comienza en la posición de memoria 100 y que sizeof(float) es 4, decir en que posición de memoria se encuentra el elemento matriz[1][2].

7. Dada la siguiente matriz:

```
struct Reg
{
    int i;
    double f;
}
typedef Reg Matriz[5][10];
Matriz dato;
```

Y sabiendo que el tamaño de un double son 8 bytes, el tamaño de un entero son 4 bytes y que la variable dato comienza en la posición de memoria 1000, calcular en que posición de memoria está el real `dato[3][6].f`.