

**Problema 1** Escribir un programa que realice el recuento de votos de una elección con 10 candidatos. Los votos están grabados en "votos.txt" y cada línea contiene un voto (Nº entero entre 1 y 10). Los resultados deben entregarse de la siguiente manera:

Candidato	Votos	%
1	xx...x	xx.xx
2	xx...x	xx.xx
...		
10	xx...x	xx.xx
total	xx...x	100.00

**Solución 1.** Con 10 variables enteras (v1, v2, ..., v10) para contar los votos de cada candidato.

```
//inicializar 10 variables (contadores)
int v1=0, v2=0, v3=0, v4=0, v5=0, v6=0,
    v7=0, v8=0, v9=0, v10=0, otros=0;

//leer todas las lineas del archivo
BR a=new BR(new FileReader("votos.txt"));
String linea;
while( (linea=a.readLine()) != null )
{
    //recuperar N° del candidato
    int nc = Integer.parseInt(linea);
```

```
//incrementar contador correspondiente
switch(nc){
case 1: ++v1; break;
case 2: ++v2; break;
case 3: ++v3; break;
case 4: ++v4; break;
case 5: ++v5; break;
case 6: ++v6; break;
case 7: ++v7; break;
case 8: ++v8; break;
case 9: ++v9; break;
case 10:++v10; break;
default:++otros;
}
}
```

```
//calcular y mostrar resultados
int total = v1+v2+v3+v4+v5+v6+v7+v8+v9+v10+otros;
double factor = 100.0/total;
U.println("candidato\tvotos\t%");
U.println("1\t"+ v1 + "\t" + v1*factor);
U.println("2\t"+ v2 + "\t" + v2*factor);
U.println("3\t"+ v3 + "\t" + v3*factor);
U.println("4\t"+ v4 + "\t" + v4*factor);
U.println("5\t"+ v5 + "\t" + v5*factor);
U.println("6\t"+ v6 + "\t" + v6*factor);
U.println("7\t"+ v7 + "\t" + v7*factor);
U.println("8\t"+ v8 + "\t" + v8*factor);
U.println("9\t"+ v9 + "\t" + v9*factor);
U.println("10\t"+v10+ "\t"+v10*factor);
U.println("total\t" + total + "\t100.0");
```

Nota. \t: tab, tabulador  
avanza hasta siguiente posición predefinida en la línea (externamente)

**Solución 2.** Con un arreglo (array) de 10 variables enteras

```
//inicializar arreglo de 10 variables enteras
final int N=10;
int[] v = new int[N];
for(int i=0; i<N; ++i) v[i]=0;
//leer todas las lineas del archivo
BR a=new BR(new FileReader("votos.txt"));
String linea; int otros=0;
while( (linea=a.readLine()) != null )
{
    //recuperar N° del candidato
    int nc = Integer.parseInt(linea);
    //incrementar contador correspondiente
    if( 1<=nc && nc<=N )
        ++v[nc-1];
    else
        ++otros;
}
}
```

**//calcular y mostrar resultados**

```
//suma de votos
int total = otros;
for(int i=0; i<N; ++i)
    total += v[i];

//mostrar resultados
double factor = 100.0/total;
U.println("candidato\tvotos\t%");
for(int i=0; i<N; ++i)
    U.println(
        (i+1)+ "\t" + //candidato
        v[i] + "\t" + //votos
        v[i]*factor); //porcentaje

U.println("total\t"+total+"\t100.0")
```

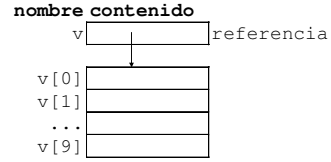
**Declaración y creación de un arreglo**

**sintaxis:** tipo[] nombre=new tipo[exp];  
 ¿exp? expresión entera que indica N° de variables del arreglo

**semántica**

1. Calcular N° de elementos (evaluar expresión).  
 Si <0, error (ArraySizeException)
2. Crear arreglo ("objeto") de N° variables del tipo indicado:  
 nombre[0], ..., nombre[N°-1]

**ejemplo:** int[] v = new int[10];



v: arreglo de variables enteras

**Indexación (acceso a una variable del arreglo)**

**ejemplos:** v[i], v[nc-1]

**sintaxis:** nombre-arreglo[índice]  
 ¿índice? expresión entera

**semántica**

1. Calcular índice (evaluar expresión)
2. Si  $0 \leq \text{índice} < N^\circ$  de elementos,  
 entonces acceder a variable[índice],  
 sino error (ArrayIndexOutOfBoundsException)

**Arreglos de objetos**

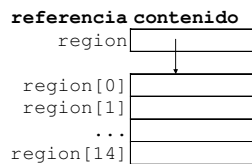
**sintaxis:** Clase[] nombre=new Clase[exp];  
 ¿exp? expresión entera que indica N° de "objetos" (referencias) del arreglo

**semántica**

1. Calcular N° de elementos (evaluar expresión).  
 Si <0, error (ArraySizeException)
2. Crear arreglo ("objeto") de referencias a objetos de la clase indicada: nombre[0], ..., nombre[N°-1]

**Ejemplo: Arreglos de Strings**

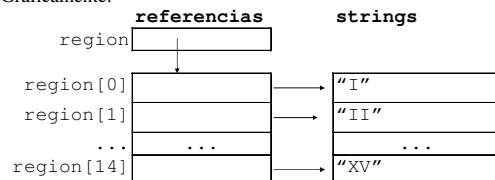
```
//arreglo para 15 strings
final int N=15;//N° máximo de strings
String[]region = new String[N];
```



region : referencia a arreglo de referencias a strings

```
//asignación
region[0]="I"; ...; region[14]="XV";
//inicialización
String[]region={"I","II","III",...,"XIV","XV"};
//uso
for(int i=0; i<N; ++i)
    U.println(region[i]);
```

Graficamente:



## Clase 15: arreglos

### Ejercicio clase 15 . Programar el sgte diálogo:

```
Ingresar habitantes de cada región
I? n°
II? n°
...
XV? n°

Porcentajes
I = xx.xx %
II = xx.xx %
...
XV = xx.xx %
```

```
//definir arreglos
final int N=15; //n° de regiones
int[]población=new int[N];
String[]region={"I","II",...,"XV"};

//obtener población de cada region y total pais
int total=0;
for(int i=0; i<N; ++i)
{
    población[i] = U.readInt(region[i] + "?");
    total += poblacion[i];
}

//calcular y mostrar porcentajes
U.println("Porcentajes");
for(int i=0; i<N; ++i)
    U.println(region[i]+"="+
        100.0*poblacion[i]/total+"%");
```

```
//definir arreglos
final int N=15; //n° de regiones
int[]población=new int[N];
String[]region={"I","II",...,"XV"};

//obtener población de cada region
for(int i=0; i<N; ++i)
    población[i]=U.readInt(region[i] + "?");

//calcular población total
int total=0;
for(int i=0; i<N; ++i)
    total += poblacion[i];

//calcular y mostrar porcentajes
U.println("Porcentajes");
for(int i=0; i<N; ++i)
    U.println(region[i]+"="+
        100.0*poblacion[i]/total+"%");
```

```
//definir arreglos
final int N=15; //n° de regiones
int[]población=new int[N];
String[]region={"I","II",...,"XV"};

//obtener población de cada region
for(int i=0; i<N; ++i)
    población[i]=U.readInt(region[i] + "?");

//calcular población total
int total=suma(poblacion,N);

//calcular y mostrar porcentajes
U.println("Porcentajes");
for(int i=0; i<N; ++i)
    U.println(region[i]+"="+
        100.0*poblacion[i]/suma+"%");
```

```
//suma primeros n elementos de un arreglo de int
static public int suma(int[]x,int n){
    int s=0;
    for(int i=0; i<n; ++i)
        s += x[i];
    return s;
}
```

#### invocación

```
int total=suma(poblacion,N);
```

- x=población  
referencia población se copia en referencia x
- n=N  
N se copia en n
- se ejecutan instrucciones de función suma

```
//suma todos los elementos de un arreglo de int
static public int suma(int[]x){
    int s=0;
    for(int i=0; i<x.length; ++i)
        s += x[i];
    return s;
}
```

**Nota.** x.length representa n° de elementos del arreglo

#### Invocación

```
int total=suma(poblacion);
```