

Taller 7

- Memoria dinámica

Memoria Dinámica

Completar para que los tres códigos sean iguales (generen un arreglo de 10 posiciones en 0)

```
int * arr;
```

1. `int i;`

```
arr = malloc(TAM*sizeof (int));
```

```
.....
```

```
.....
```

```
mostrarArreglo(arr,TAM);
```

```
free(arr);
```

2.

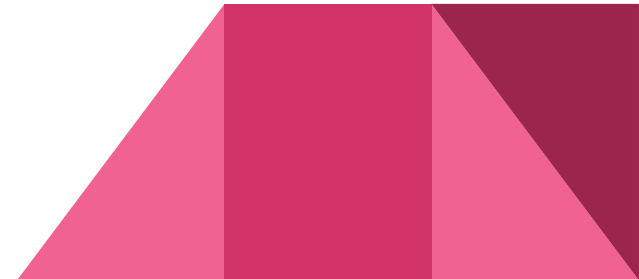
```
.....
```

```
arr = realloc(arr, .....);
```

```
.....
```

```
.....
```

3. `arr = calloc(.....,);`



Arreglo dinámico de enteros.

Hacer una función que lea números enteros de entrada estándar (los lee con `getint`) y los guarde en un arreglo. El arreglo debe crecer a medida que se necesita. La carga en el arreglo finaliza cuando un valor se repite por primera vez.

Tener en cuenta:

- Modularizar cuando se pueda
- Fijarse en qué puntos del código hay que reasignar memoria

```
int main(void) {
    int dim;
    int *arr;
    arr = cargarNumeros(&dim);

    for (int i = 0; i < dim; i++)
        printf("%d ", arr[i]);

    free(arr);
    return 0;
}
```

El supervector

Implementar un **programa** que lea de la entrada estándar hasta EOF y genere el *supervector* que todo lo puede guardar. Luego imprimir el contenido del vector en salida estándar.

