

En esta práctica se pueden usar los **elementos de C permitidos** en prácticas anteriores, y hay que seguir teniendo en cuenta las **restricciones** de funcionamiento de esas prácticas.

Esta práctica está diseñada para seguir ejercitando el uso de **arrays**.

De momento, **no está permitido** el uso de strings, u otros elementos del lenguaje C.

Se deben **usar todos los prototipos** indicados y sólo esos prototipos.

En C se pueden generar números aleatorios, incluyendo dos bibliotecas: "stdlib.h" y "time.h". Lo primero es inicializar el proceso de generación, mediante srand(time(NULL)). Lo segundo, usar rand() para generar un número aleatorio concreto. Si queremos que ese valor aleatorio se genere en el intervalo [0,**rango**-1], usaremos rand()%**rango**.

En la práctica actual, se deben generar aleatoriamente los DNIs, calculándose la letra, para a continuación printarlos. Por último, para las letras de los DNIs, se deben calcular las frecuencias absolutas y relativas, printarlas y dibujar su diagrama de barras.

La función *letra_calculada (unsigned)* se continuará definiendo con un array de chars (sin if ni switch). En esta ocasión, **no** se utilizará **asignación dinámica de memoria** (malloc, free).

```
> DNIs y letras:
55503358-B 9154144-Y 1762653-W 63874635-R 37310776-S 21241350-E 50485270-V 96887804-M
46232223-S 56340330-J 70345255-P 22707650-A 86489468-S 97955697-P 51378395-Y 38176864-S
13647673-W 49209690-V 86339671-V 76142970-K 18991029-K 77977859-Q 87962423-G 66974655-N
50539777-Z 59695687-S 73363662-X 41780254-H 47900943-Q 30263934-M 25037556-D 42302497-T
43088747-H 50059578-D 90981995-K 21037244-H 75861867-R 17066311-N 67533757-F 87958659-N
97782065-A 78791697-E 83415697-X 10644302-V 14435874-Q 90000137-X 31929185-X 35214960-M
27899215-P 65127389-Z 89239178-Y 92403339-B 67278598-X 47810610-G 36204752-S 76092429-B
97194106-Q 33944189-F 41481606-W 14578945-G 59122563-M 13247264-T 1232926-B 68988513-J
16482625-C 58477883-S 11949758-Q 21984625-Y 50562796-X 97091922-K 76558273-J 35476765-R
39721178-V 51724593-P 39730344-Y 63610038-L 4019499-L 7440229-M 37062197-C 85817090-L
75116542-Z 76301179-J 59850151-B 90103743-R 86569209-S 29892273-R 29498476-X 57128450-S
57804205-F 38517594-T 2370438-N 65081687-J 45734907-M 12518312-X 45562403-R 87963672-B
56707293-B 89595873-V 49583245-Y 22500157-Q

> frecuencias absolutas, relativas, y diagrama de barras:
A 2 0.02 **
B 7 0.07 **
C 2 0.02 **
D 2 0.02 **
E 2 0.02 **
F 3 0.03 ***
G 3 0.03 ***
H 3 0.03 ***
J 5 0.05 *****
K 4 0.04 ****
L 3 0.03 ***
M 6 0.06 *****
N 4 0.04 ****
P 4 0.04 ****
Q 6 0.06 *****
R 6 0.06 *****
S 9 0.09 *****
T 3 0.03 ***
V 6 0.06 *****
W 3 0.03 ***
X 8 0.08 *****
Y 6 0.06 *****
Z 3 0.03 ***
```

Figura 1. Ejemplo de ejecución del programa

//includes

```
#include "stdio.h"
#include "time.h" // time()
#include "stdlib.h" // srand(), rand()
```

//defines

```
#define N 100
```

//prototipos usados en practicas anteriores

```
unsigned resto_DNI(unsigned );
char letra_calculada(unsigned );
```

//prototipos de esta práctica

```
void rand_dig (char *);
void rand_DNI (unsigned *);
void rand_DNIs (unsigned [N], char [N]);
void print_DNIs (unsigned [N], char [N]);
void calcular_frecuencias (char [N], unsigned [26], float [26]);
void print_frecuencias (unsigned [26], float [26]);
void print_barra (float);
```

//main

```
int main()
{
    unsigned ArrayDNIS[N],Frecuencias_Absolutas[23] = {0};
    char LetrasDNIS[N];
    float Frecuencias_Relativas [23] = {0};

    srand(time(NULL));
    rand_DNIs(ArrayDNIS, LetrasDNIS);
    calcular_frecuencias(LetrasDNIS,Frecuencias_Absolutas,Frecuencias_Relativas);
    print_frecuencias(Frecuencias_Absolutas,Frecuencias_Relativas);
    return 0;
}
```

// definición de las funciones

```
void rand_dig(char* dig) {
    *dig = '0' + (rand() % 10);
}
```

```
void rand_DNI(unsigned* DNI) {
    *DNI = 0;
    char digito;
    for (int i = 1; i <= 8; i++) {
        rand_dig(&digito);
```

```
        *DNI = *DNI * 10 + (digito - '0');
    }
}

void rand_DNIs(unsigned ArrayDNIS[N], char LetrasDNIS[N]) {
    unsigned DNI = 0;
    int resto, letraCalculada;

    for (int i = 0; i < N; i++) {
        rand_DNI(&DNI);
        ArrayDNIS[i] = DNI;
        resto = resto_DNI(DNI);
        letraCalculada = letra_calculada(resto);
        LetrasDNIS[i] = letraCalculada;
    }
    print_DNIs(ArrayDNIS, LetrasDNIS);
}

unsigned resto_DNI(unsigned DNI) {
    return DNI % 23;
}

char letra_calculada(unsigned resto) {
    char letraCalculadaDNI[] = {'T', 'R', 'W', 'A', 'G', 'M', 'Y', 'F', 'P', 'D', 'X',
                                'B', 'N', 'J', 'Z', 'S', 'Q', 'V', 'H', 'L', 'C', 'K', 'E'};
    return letraCalculadaDNI[resto];
}

void print_DNIs(unsigned ArrayDNIS[N], char LetrasDNIS[N]) {
    printf(" >DNIs Y Letras: \n");
    for (int i = 0; i < N; i++) {
        printf("%08u - %-10c", ArrayDNIS[i], LetrasDNIS[i]);
    }
}

void calcular_frecuencias(char LetrasDNIS[N], unsigned Absolutas[23], float
Relativas[23]) {

    for (int i = 0; i < N; i++) {
        if (LetrasDNIS[i] >= 'A' && LetrasDNIS[i] <= 'W') {
            Absolutas[LetrasDNIS[i] - 'A']++;
        }
    }

    for (int i = 0; i < 23; i++) {
        Relativas[i] = (float)Absolutas[i] / N * 100;
    }
}

void print_barra(float barras) {
    int veces = barras;
```

```
    for (int i = 0; i < veces; i++) {  
        printf("*");  
    }  
}  
  
void print_frecuencias(unsigned Absoluto[23], float Relativo[23]) {  
    char letras[23] = {'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L',  
        'M', 'N', 'P', 'Q', 'R', 'S', 'T', 'V', 'W', 'X', 'Y', 'Z'};  
    printf("\n > frecuencias absolutas, relativa y diagramas de barra:\n");  
    for (int i = 0; i < 23; i++) {  
        printf("%c %2u %.2f ", letras[i], Absoluto[i], Relativo[i]/100);  
        print_barra(Relativo[i]);  
        printf("\n");  
    }  
}
```