

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 ejercitar el uso de **struct**.

Se deben **usar todos y cada uno de los prototipos** indicados en esta práctica, y los de prácticas anteriores que se requieran.

En esta ocasión, no se utilizará **asignación dinámica de memoria** en ninguna función.

En esta práctica, con las estructuras dadas más abajo, se deben generar aleatoriamente los 80 alumnos de una clase, printando al final lo obtenido. El año de nacimiento debe estar comprendido entre 1999 y 2005. La fecha de nacimiento debe ser válida, considerando además si el año es bisiesto.

74595571Q	hernandez, jose	01-07-2001
63028652G	sanchez, pilar	25-05-2000
09231432Z	gutierrez, roberto	25-04-2000
86382964R	lopez, roberto	01-06-2004
51860434B	lopez, roberto	16-10-2005
64302031B	gutierrez, fernando	11-01-1999
62261273C	lopez, pilar	20-08-1999
53398705L	perez, jose	13-04-2003
81943759A	hernandez, raquel	15-01-2005
02239248Z	fernandez, sandra	15-12-2002
79735689W	perez, raquel	31-05-2001
13247363F	lopez, fernando	17-10-2002
31510319C	hernandez, raquel	26-10-2002
07303129P	perez, sandra	27-10-2000
14345698T	garcia, roberto	26-05-2000
27107190B	fernandez, raquel	05-06-1999
75560462B	perez, sandra	15-01-2000
72467804W	hernandez, isabel	27-01-1999
01288263X	lopez, roberto	13-07-1999
90660497V	fernandez, roberto	03-01-2005
70258250N	fernandez, isabel	27-03-1999
69482074V	hernandez, mario	17-11-2005
15172093M	lopez, roberto	27-10-2003
43222027J	hernandez, fernando	06-07-1999
65708224F	perez, jose	26-03-2002
24443618T	lopez, roberto	31-11-1999
10367351D	perez, pilar	14-10-2000
91025003L	gutierrez, pilar	24-08-2001
88601090F	sanchez, pilar	30-06-2005
68874524N	garcia, jose	31-01-2000

Figura 1. Ejemplo de ejecución del programa

```
//include
```

```
//define
```

```
#define A 80
```

```
//struct
```

```
typedef struct
```

```
{ unsigned dia, mes, anyo;
} DATE;
```

```
struct ALUMNO
{ char DNI[9+1];
  char nom_comp [20+1];
  DATE nac;
};
```

```
//const
```

```
const char nombres[8][10+1]={"jose","roberto","fernando","mario",
"sandra", "raquel", "isabel", "pilar"};
const char apellidos[8][10+1]={"garcia", "hernandez", "gutierrez",
"gomez", "fernandez", "lopez", "sanchez", "perez" };
```

```
//prototipos usados en practicas anteriores
```

```
unsigned resto_DNI(unsigned );
char letra_calculada(unsigned );
void rand_str_DNI(char [10]);
```

```
//prototipos de esta práctica
```

```
void print_DATE(DATE );
void print_ALUMNO (ALUMNO);
void rand_DATE(DATE *);
void rand_nom_comp(char [20+1]);
void rand_ALUMNO(ALUMNO *);
unsigned es_fecha_valida(DATE);
```

```
//main
```

```
int main()
{ ALUMNO alumno;
  int i;
  srand(time(NULL));
  for(i=0;i<A;i++){
    rand_ALUMNO(&alumno);
    print_ALUMNO(alumno);
  }
  return 0;
}
```

## // definición de las funciones

```
unsigned es_fecha_valida (DATE fecha){
```

```
    switch (fecha.mes) {
```

```
        case 1: fecha.dia = rand()%32+1;
                break;
```

```
        case 2: if(!fecha.anyo % 4){
                fecha.dia = rand ()%30+1;
                }else{
                fecha.dia = rand()%29+1;
                }
                break;
```

```
        case 3: fecha.dia = rand()%32+1;
                break;
```

```
        case 4: fecha.dia = rand()%31+1;
                break;
```

```
        case 5: fecha.dia = rand()%32+1;
                break;
```

```
        case 6: fecha.dia = rand()%31+1;
                break;
```

```
        case 7: fecha.dia = rand()%32+1;
                break;
```

```
        case 8: fecha.dia = rand()%32+1;
                break;
```

```
        case 9: fecha.dia = rand()%31+1;
                break;
```

```
        case 10: fecha.dia = rand()%32+1;
                break;
```

```
        case 11: fecha.dia = rand()%31+1;
                break;
```

```
        case 12: fecha.dia = rand()%32+1;
                break;
```

```
    }
    return fecha.dia;
```

```
}
```

```
void rand_DATE (DATE *fecha){
    fecha->anyo = rand()%7 + 1999;
    fecha->mes = rand()%12 + 1;
    fecha->dia = es_fecha_valida(*fecha);
}
```

```
void rand_nom_comp(char nom_comp[23]){
    const char nombres[8][10+1]={ "jose", "roberto", "fernando", "mario", "sandra", "raquel",
    "isabel", "pilar"};
    const char apellidos[8][10+1]={ "garcia", "hernandez", "gutierrez", "gomez",
    "fernandez", "lopez", "sanchez", "perez" };
    char delimitador[3] = " ";
```

```
strcpy(nom_comp, apellidos[rand()%8]);
strcat(nom_comp, delimitador);
strcat(nom_comp, nombres[rand()%8]);
}

void rand_str_DNI(char dni[10]){
    int i, num;
    for(i=0;i<8;i++) //Obtenemos el Dni aleatorio en un string
        dni[i]=rand()%10 + '0';
    for(i=0, num = 0;i<8;i++) //Pasamos el dni en string a un tipo int para calcular la letra
        num= (num*10) + (dni[i] - '0');
    dni[i] = letra_calculada(resto_DNI(num)); //Calculamos la letra
    dni[i+1] = '\0';
}

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

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

void rand_ALUMNO(ALUMNO *alumno){
    char dni[10], nom_completo[23];
    DATE fecha;

    rand_str_DNI(dni);
    strcpy(alumno->DNI, dni);
    rand_nom_comp(nom_completo);
    strcpy(alumno->nom_comp, nom_completo);
    rand_DATE(&fecha);
    alumno->nac = fecha;
}

void print_DATE(DATE fecha){
    printf("%u-%u-%u\n", fecha.dia, fecha.mes, fecha.anyo);
}

void print_ALUMNO(ALUMNO alumno){
    printf("%s %s ", alumno.DNI, alumno.nom_comp);
    print_DATE(alumno.nac);
}
```