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.

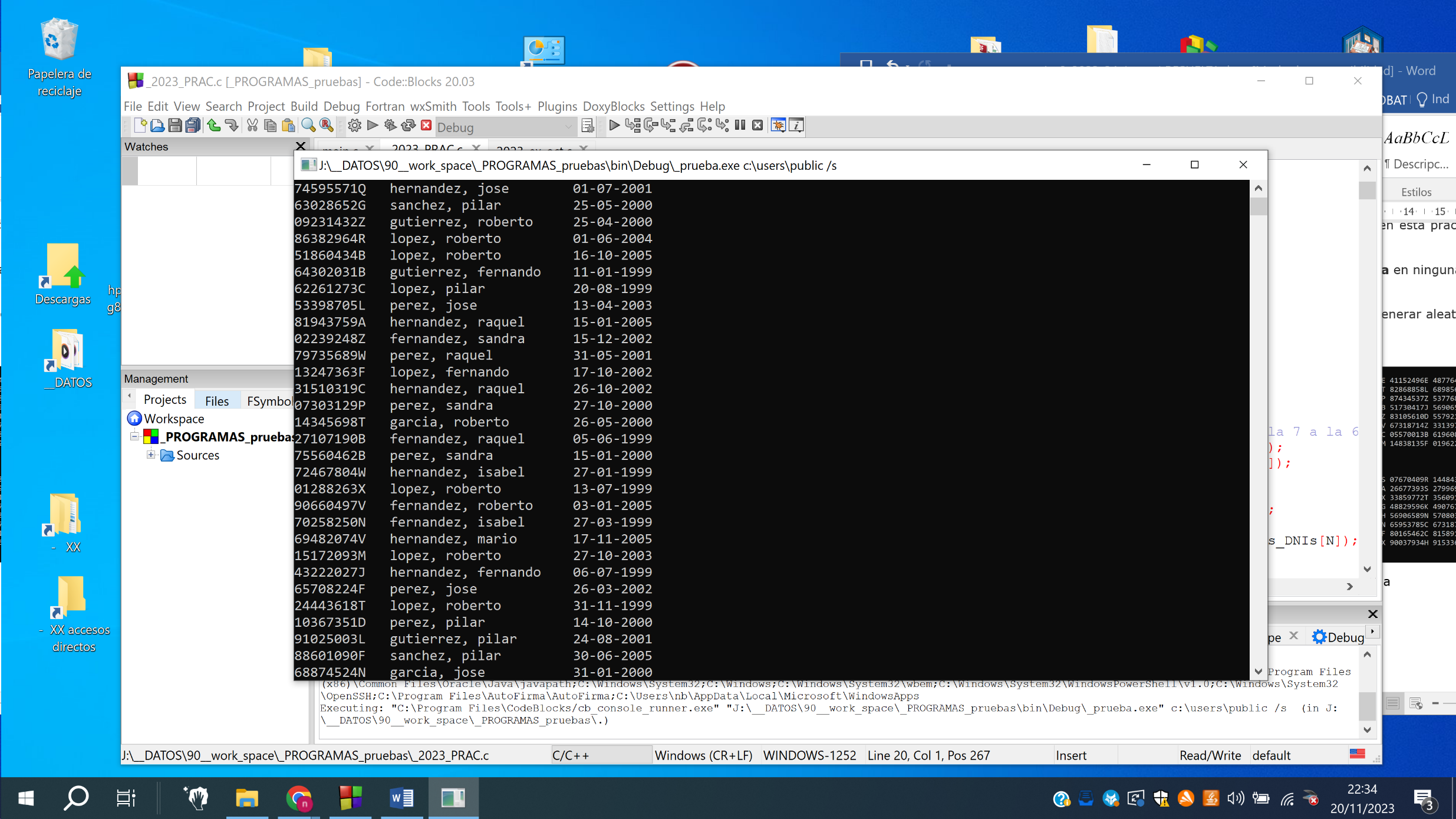


Figura 1. Ejemplo de ejecución del programa

//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 letra[23] ={'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);

}