ESTUDIANTE ESTUDIANTE EXAMEN EXTRAORDINARIO 8 de julio de 2022												UPM	0					
APELLIDOS (en MAYÚSCULAS)																		
NOMBRE (en MAYÚSCULAS)																		

1.- (3 ptos) Escribir un programa que lea de teclado letras mayúsculas (cualquier otro carácter introducido no se tendrá en cuenta) hasta que aparezca la Z. Después deberá imprimir, para las mayúsculas que hayan aparecido, cuántas veces lo han hecho y su porcentaje frente al resto, si una letra no ha aparecido no se deberá mostrar. Un ejemplo de ejecución puede ser:

```
Introduzca letras mayusculas (Z para acabar):

gAAATCnVVQeJJJZ

Las veces que ha aparecido cada letra y su secuencia es:

A 3 23.08
C 1 7.69
J 4 30.77
Q 1 7.69
I 1 7.69
U 2 15.38
Z 1 7.69
Process returned 16 (0x10) execution time: 34.365 s
Press any key to continue.
```

```
void main()
{ int total=0, letras[26]={0}, i;
    char car;

printf("Introduzca letras mayusculas (Z para acabar):\n");

do
    { do
      { scanf("%c", &car);
      } while (car<'A' | | car>'Z');

    total++;
    letras[car-'A']++;

} while (car!='Z');

printf ("\n\nLas veces que ha aparecido cada letra y su secuencia es:\n");

for (i=0; i<26; i++)
    if (letras[i])
      printf ("\n%c %4d %5.2f", i+'A', letras[i], letras[i]/(float)total*100);
}</pre>
```

2.- (2 ptos) Escribir una función que, recibiendo como parámetros una matriz de NxN enteros (siendo N una cte ya definida) y dos enteros f1 y f2 que representan dos filas válidas de la matriz (son valores comprendidos entre 0 y N-1 y el primero es menor o igual al segundo), ponga a 0 los 8 bits de más a la izda de todos los números que estén en esas filas. NO SE PODRÁ USAR OTRA MATRIZ AUXILIAR.

```
void recuadrar (int mat[N][N], unsigned f1, unsigned f2)
{ int i, j;

for (i=f1; i<=f2; i++)
    for (j=0; j<N; j++)
      mat[i][j] = (mat[i][j]<<8)>>8;
}
```

3.- (2 ptos) Escribir una función que, recibiendo como parámetros dos ficheros de texto abiertos y una letra (mayúscula o minúscula), lea del primer fichero palabras y grabe en el segundo aquellas que empiecen por la letra pasada como parámetro. El primer fichero está formado por líneas que contienen palabras (de un máximo de 15 caracteres) y espacios en blanco entre ellas.

```
void buscar_pal (FILE *f_in, FILE *f_out, char letra)
{ char palabra[16];

while (fscanf(f_in, "%s", palabra) != EOF)
  if (letra == palabra[0])
    fprintf(f_out, "%s ", palabra);
}
```

4.- (3 ptos) Escribir un programa genere combinaciones simples válidas del juego de la primitiva y las almacene en un fichero cuyo nombre se leerá por teclado. Cada combinación a generar va a consistir en 6 números distintos entre 1 y 49 y un número de reintegro entre 0 y 9, es decir, 7 valores. Una vez generada la combinación se deberá guardar en el fichero. Antes de empezar a generarlas se deberá introducir por teclado cuántas combinaciones hay que generar, no permitiendo que se introduzca un número negativo para este dato. Por ejemplo, una combinación válida sería:

8	41	27	15	18	31	0	_
---	----	----	----	----	----	---	---

donde los 6 primeros números corresponden a la combinación y el último es el reintegro.

```
int main ()
{ FILE *f_out;
  unsigned combi[7], num;
  int i, j, k;
  char nom_fich[50];

srand(time(NULL));

printf ("Introduzca el nombre del fichero de combinaciones: ");
  gets(nom_fich);

if (! (f_out = fopen (nom_fich, "wb")))
  { printf ("\n\nError en la apertura del fichero %s.\n", nom_fich);
    return (1);
  }

printf ("\n¿Cuántas combinaciones se deben generar?: ");
  scanf("%u", &num);
```

IDENTIFICADOR ESTUDIANTE POLITÉCNICA	FL	JNDAN EX	ΑM	ΕN	EX	TR	40	 NΑ	 	ÓN		
APELLIDOS (en MAYÚSCULAS)												

_0	-
1)
UPM	

APELLIDOS (en MAYÚSCULAS)																	
NOMBRE (en MAYÚSCULAS)																	
		<u> </u>	<u> </u>						_								
for (i=0; i <num; combinaciones<="" generar="" i++)="" las="" para="" th="" todas=""></num;>																	
{ for (j=0; j<6;) //para generar una combinación { combi[j] = rand()%49+1; //generación de un valor de la combinación																	
for (k=j-1; k>=0 && combi[j] != combi[k]; k); //comprobar que el número no se ha generado antes if (k<0) //sólo si el número no coincide con los anteriores se genera otro nuevo j++; }																	
combi[j] = rand()%10; //re	_																
fwrite(combi, sizeof(combi }), 1, f_0i	ut);															
<pre>fclose(f_out); return (0); }</pre>																	