

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

--	--	--



FUNDAMENTOS DE PROGRAMACIÓN
EXAMEN EXTRAORDINARIO
8 de julio de 2022



APELLIDOS (en MAYÚSCULAS)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

NOMBRE (en MAYÚSCULAS)

--	--	--	--	--	--	--	--	--	--	--	--	--

```
for (i=0; i<num; i++) //para generar todas las combinaciones
{ 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; //reintegro
  fwrite(combi, sizeof(combi), 1, f_out);
}

fclose(f_out);
return (0);
}
```