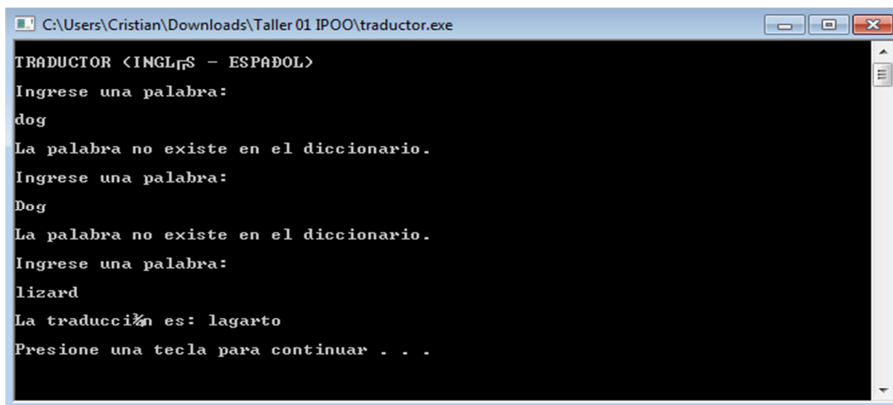


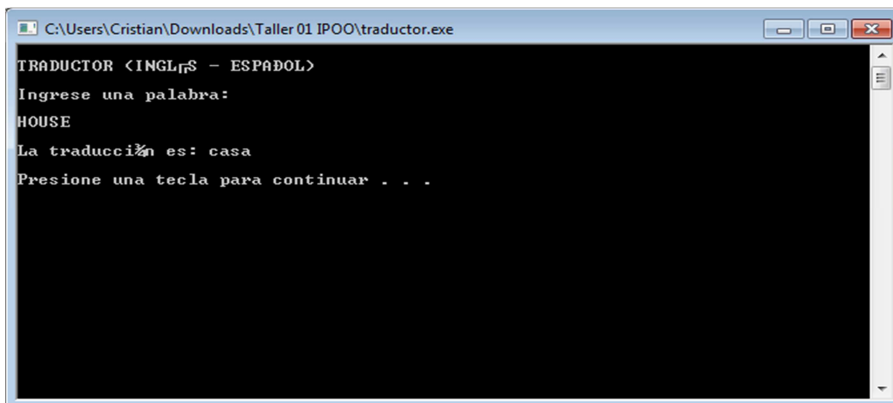
PUNTO 2: Traductor

Para este punto se usaron dos arreglos en los cuales están almacenadas 10 palabras, el primer arreglo contiene las palabras en inglés y el otro su respectiva traducción al español. El programa pide ingresar al usuario la palabra deseada, la palabra ingresada es pasada por una función que convierte a minúsculas, luego pasa por funciones que por medio de for's buscan si la palabra está en el primer arreglo, de estarlo retorna la posición y se imprime la posición del segundo arreglo, de no estarlo imprime el mensaje de que no existe la palabra.

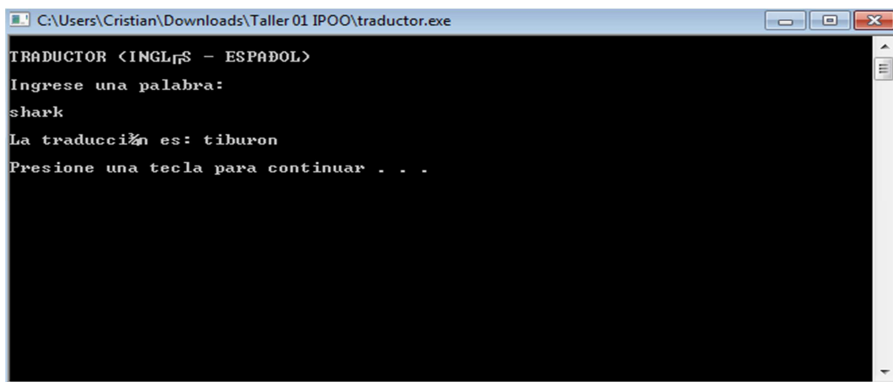
PRUEBAS:



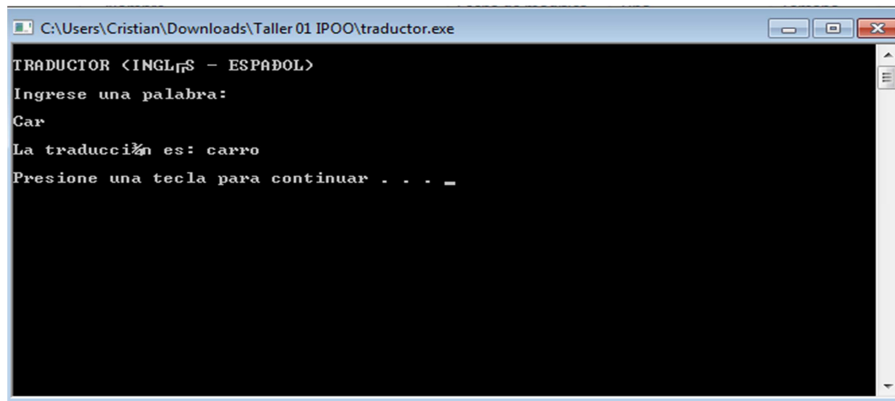
```
C:\Users\Cristian\Downloads\Taller 01 IPOO\traductor.exe
TRADUCTOR <INGLÉS - ESPAÑOL>
Ingrese una palabra:
dog
La palabra no existe en el diccionario.
Ingrese una palabra:
Dog
La palabra no existe en el diccionario.
Ingrese una palabra:
lizard
La traducción es: lagarto
Presione una tecla para continuar . . .
```



```
C:\Users\Cristian\Downloads\Taller 01 IPOO\traductor.exe
TRADUCTOR <INGLÉS - ESPAÑOL>
Ingrese una palabra:
HOUSE
La traducción es: casa
Presione una tecla para continuar . . .
```



```
C:\Users\Cristian\Downloads\Taller 01 IPOO\traductor.exe
TRADUCTOR <INGLÉS - ESPAÑOL>
Ingrese una palabra:
shark
La traducción es: tiburón
Presione una tecla para continuar . . .
```



PUNTO 3: Inventario.

Para realizar este punto se usaron dos vectores (usando la librería vectores), el primero para almacenar el artículo y el otro la cantidad.

Se imprime el menú principal y se pide la opción, esta es pasada a un switch:

En caso de ser 1(Ingresar) se recorre el vector de artículos por medio de un for para verificar si ya está y en ese caso sumarle cantidad, en caso contrario almacena el nombre en el primer vector y la cantidad en el segundo por medio de “.push_back()”.

En caso de ser 2(Eliminar) se recorre el vector de artículos por medio de un for que compara y captura la posición en la que lo encuentre, luego se elimina lo que esté en esa posición de los vectores por medio de “.begin()+(la posición capturada)” .

En caso de ser 3(Mostrar) por medio de un for se recorren los vectores y se imprime cada entrada.

En caso de ser 4(Salir) el programa termina.

PRUEBAS:

Ingresar:

```
C:\Users\Cristian\Downloads\Taller 01 IPOO\inventario.exe
*****
***** BIENVENIDO A MI INVENTARIO *****
*****
***** Seleccione una opción: *****
*****
***** 1. Ingresar un elemento. *****
***** 2. Eliminar un elemento. *****
***** 3. Mostrar elementos. *****
***** 4. Salir. *****
*****
Ingrese su opción:
1
*****
***** Ingresar un elemento: *****
*****
Ingrese el nombre del elemento:
Arroz
Ingrese la cantidad de elementos:
10
Ingreso exitoso!
*****
***** BIENVENIDO A MI INVENTARIO *****
*****
***** Seleccione una opción: *****
*****
***** 1. Ingresar un elemento. *****
***** 2. Eliminar un elemento. *****
***** 3. Mostrar elementos. *****
***** 4. Salir. *****
*****
Ingrese su opción:

```

Mostrar:

```
C:\Users\Cristian\Downloads\Taller 01 IPOO\inventario.exe
10
Ingreso exitoso!
*****
***** BIENVENIDO A MI INVENTARIO *****
*****
***** Seleccione una opción: *****
*****
***** 1. Ingresar un elemento. *****
***** 2. Eliminar un elemento. *****
***** 3. Mostrar elementos. *****
***** 4. Salir. *****
*****
Ingrese su opción:
1
*****
***** Ingresar un elemento: *****
*****
Ingrese el nombre del elemento:
Arroz
Ingrese la cantidad de elementos:
3
Ingreso exitoso!
*****
***** BIENVENIDO A MI INVENTARIO *****
*****
***** Seleccione una opción: *****
*****
***** 1. Ingresar un elemento. *****
***** 2. Eliminar un elemento. *****
***** 3. Mostrar elementos. *****
***** 4. Salir. *****
*****
Ingrese su opción:
3
*****
***** Elementos Almacenados: *****
*****
***** Nombre: ***** Cantidad: *****
***** Arroz ***** 13 *****
*****

```

Ingresar:

```
C:\Users\Cristian\Downloads\Taller 01 IPOO\inventario.exe
*****
Nombre:      Cantidad:
*****
Arroz ..... 13
*****

***** BIENVENIDO A MI INVENTARIO *****
*****
Seleccione una opción:
*****
1. Ingresar un elemento.
2. Eliminar un elemento.
3. Mostrar elementos.
4. Salir.
*****

Ingrese su opción:
1

***** Ingresar un elemento: *****
*****
Ingrese el nombre del elemento:
jabon
Ingrese la cantidad de elementos:
3
Ingreso exitoso!

***** BIENVENIDO A MI INVENTARIO *****
*****
Seleccione una opción:
*****
1. Ingresar un elemento.
2. Eliminar un elemento.
3. Mostrar elementos.
4. Salir.
*****

Ingrese su opción:
3

***** Elementos Almacenados: *****
*****
Nombre:      Cantidad:
*****
Arroz ..... 13
jabon ..... 3
*****
```

Eliminar:

```
C:\Users\Cristian\Downloads\Taller 01 IPOO\inventario.exe
*****
Nombre:      Cantidad:
*****
Arroz ..... 13
jabon ..... 3
*****

***** BIENVENIDO A MI INVENTARIO *****
*****
Seleccione una opción:
*****
1. Ingresar un elemento.
2. Eliminar un elemento.
3. Mostrar elementos.
4. Salir.
*****

Ingrese su opción:
2

***** Eliminar un elemento: *****
*****
Ingrese el nombre del elemento:
jabon
Eliminación exitosa!

***** BIENVENIDO A MI INVENTARIO *****
*****
Seleccione una opción:
*****
1. Ingresar un elemento.
2. Eliminar un elemento.
3. Mostrar elementos.
4. Salir.
*****

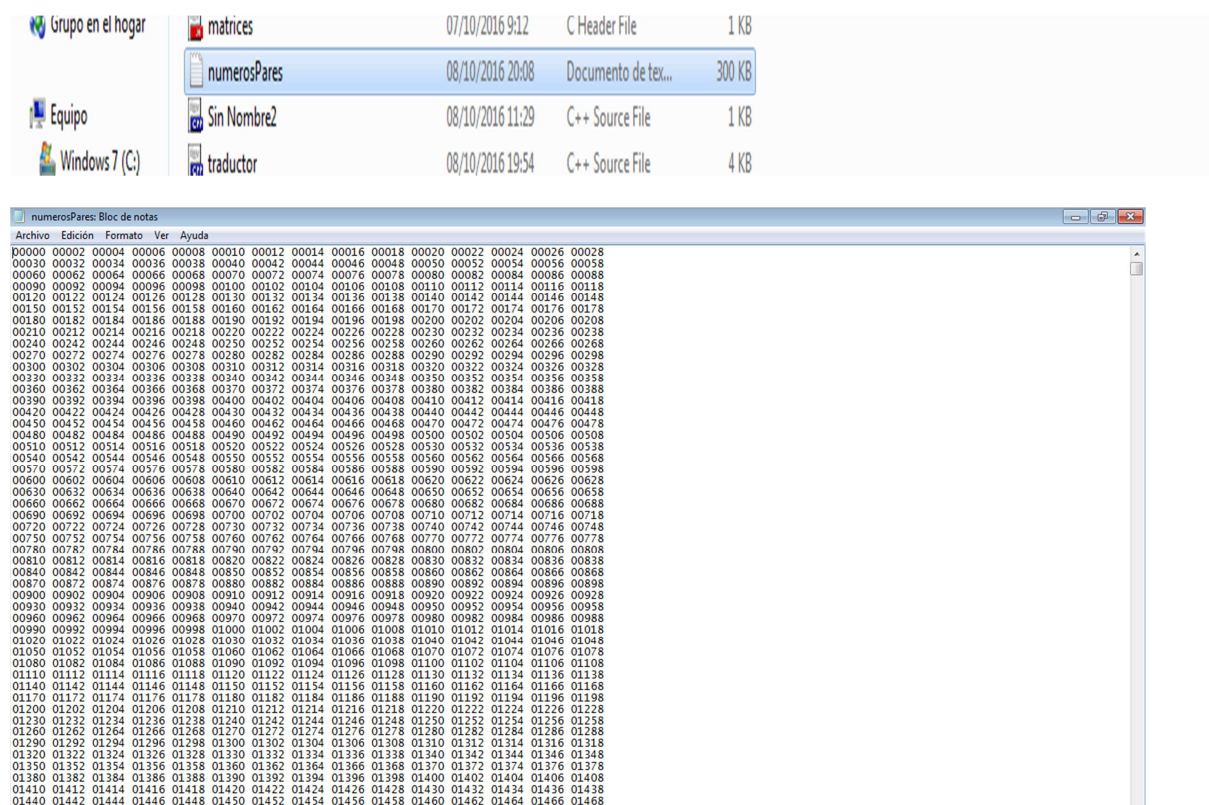
Ingrese su opción:
3

***** Elementos Almacenados: *****
*****
Nombre:      Cantidad:
*****
Arroz ..... 13
*****
```

PUNTO 4: Generar pares.

Para realizar este punto utilizamos un for con avance de 2 en 2, un contador para las columnas y abrimos un archivo(usando la librería fstream), se creó una función que determinan los ceros que se deben agregar a cada número para que todos se impriman de 5 dígitos, el contador inicia desde 0 y en cada iteración se imprime el numero pasado por la función en el archivo creado y se le suma 1 al contador, cuando el contador es igual a 15 es porque ya se han impreso 15 columnas entonces se imprime un salto de línea y se setea el contador a 0. Al final de las iteraciones del for se cierra el archivo.

PRUEBA:



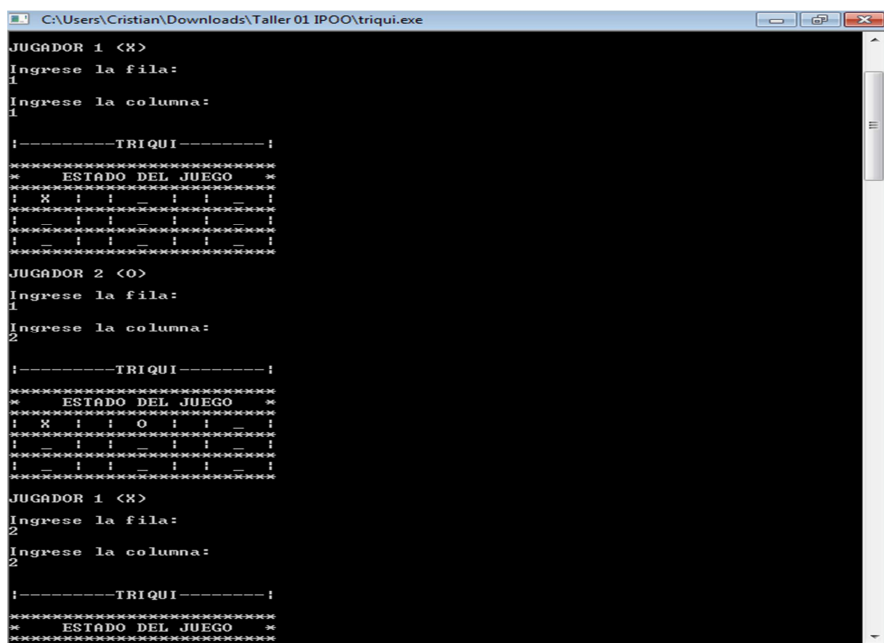
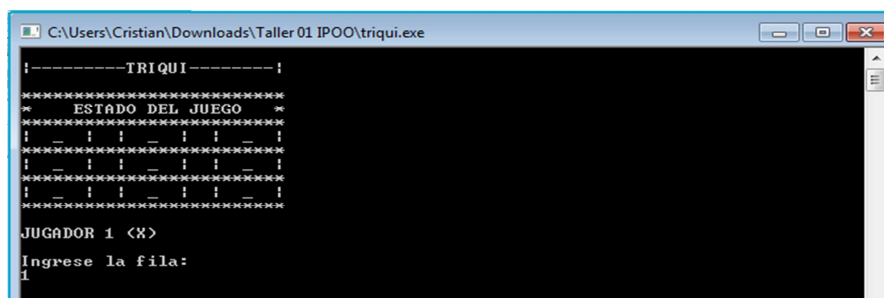
PUNTO 5: Triqui.

Para este punto se creó una matriz de 0`s y funciones:

- 1.) Por medio de un switch dibuja la matriz (si hay 1 imprime X, si hay 2 imprime O, en otro caso imprime _).
- 2.) Según la posición ingresada verifica si está ocupada, una función que valida que se introduzca una posición correcta.
- 3.) Valida el ganador, verificando si las entradas de una fila, columna o diagonal son iguales e imprime quién gana o si hay empate (para lo cual se definió un contador que aumenta en cada jugada si es igual a 9 y no hay ganador es porque hay empate).

Como las filas y las columnas se numeran de 1-3 cuando el usuario ingresa la posición se le resta 1 para poder usar correctamente las posiciones de la matriz.

PRUEBAS:

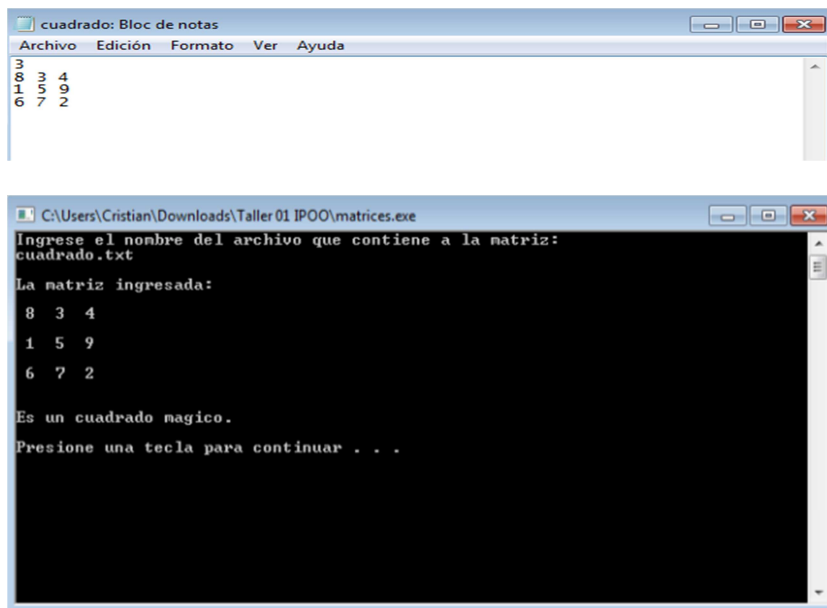


PUNTO 6: Matrices.

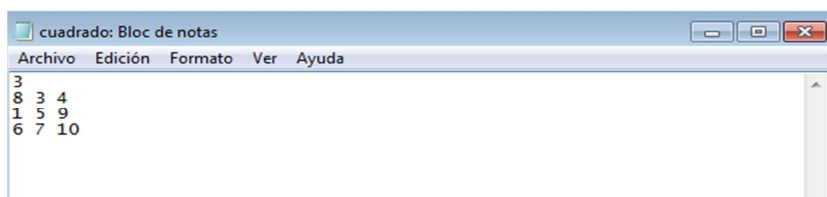
Por medio de la librería fstream se lee el archivo que contiene el cuadrado carácter a carácter y se almacena cada carácter en un vector, se toma la primera posición del vector como el tamaño de la matriz y se elimina, con la ayuda de punteros se crea la matriz del tamaño tomado, por medio de un for se pasa cada valor que contiene el vector a la matriz, luego se determina el valor que debe tener la suma de cada lado de la matriz, finalmente por medio de funciones que usan for's para recorrer la matriz e if's se verifica que los elementos de la matriz sean únicos, estén entre 1 y n cuadrado (siendo n el tamaño), y la suma de sus dimensiones sean iguales e iguales a el valor determinado, en caso de cumplir con lo anterior con ayuda de un for se imprime la matriz y un mensaje diciendo que es un cuadrado mágico o en caso de no cumplir se imprime la matriz y el mensaje de que no es un cuadrado mágico.

PRUEBAS:

Con cuadrado mágico:



Con cuadrado no mágico



```
C:\Users\Cristian\Downloads\Taller01 IPCC\matrices.exe
Ingrese el nombre del archivo que contiene a la matriz:
cuadrado.txt
La matriz ingresada:
8 3 4
1 5 9
6 7 10
No es un cuadrado magico.
Presione una tecla para continuar . . . _
```