

## Laboratorio 2 (Dirigido)

### Instrucciones

Se ha creado la tarea “Laboratorio 02 (Dirigido)” en la sección Laboratorio Dirigido 2 de la página del curso-horario en el PAIDEIA. Cada alumno deberá subir un archivo comprimido (formato ZIP) con todos los archivos del desarrollo realizado en el laboratorio. El nombre del archivo comprimido deberá tener el formato: L2\_<Código del alumno de 8 dígitos>.zip

Como ejemplo, el nombre del archivo de alumno 20186969 del horario 06M2 sería “L2\_20186969.zip”

Es COMPLETA RESPONSABILIDAD DE CADA ALUMNO el colocar los nombres correctos a sus archivos y evitar así confusiones al momento de la calificación.

### Actividad 1

- Explicación del concepto de arreglos unidimensionales
  - Qué es un arreglo unidimensional (vector)
  - Cómo se define un arreglo unidimensional en C++
  - Acceso a elementos de un arreglo unidimensional en C++
  - Paso de un arreglo unidimensional como parámetro de una función en C++.
- Explicación del concepto de arreglos bidimensionales
  - Qué es un arreglo bidimensional (matriz)
  - Cómo se define un arreglo bidimensional en C++
  - Acceso a elementos de un arreglo bidimensional en C++
  - Paso de un arreglo bidimensional como parámetro de una función en C++.

### Actividad 2

Resolución del siguiente problema, utilizando arreglos unidimensionales y bidimensionales, funciones, selectivas e iterativas en conjunto con los alumnos.

#### **El juego de la Sopa de Letras o Pupiletras**

La creación de estos puzzles es atribuida oficialmente a Norman E. Gibat, un estadounidense de Norman, Oklahoma. En 1968, Gibat buscaba una forma de atraer más lectores para su publicación Selenby Digest cuando inventó el juego que ahora conocemos como Sopa de Letras.

Aunque, la invención de los puzzles de Sopa de Letras es también reivindicada por Pedro Ocón de Oro, un madrileño a quien se le reconoce la creación de más de 125 pasatiempos en español. Esta incertidumbre hizo que la historia de las Sopas de Letras se tumbara en favor de Gibat y le diera el título de creador oficial. Sin embargo, es generalmente aceptada la hipótesis de que Gibat se inspirara en la versión española para crear sus puzzles.

Como ni Norman E. Gibat ni Ocón de Oro patentaron sus creaciones, los puzzles de Sopa de Letras se han difundido muy rápidamente por diversas publicaciones de todo el mundo, sin que exista una denominación común y universal para el juego. Gibat y Ocón de Oro crearon la versión clásica de este juego, en la que el jugador debe encontrar en una rejilla de letras todas las palabras de una lista temática. El tamaño y formato del tablero ha sido siempre flexible para acomodar los diferentes niveles de dificultad.

Nosotros realizaremos en esta ocasión nuestro juego PUPI-LPOO, el cual se basará en el tradicional juego de Sopa de Letras o también llamado Pupiletras con las siguientes consideraciones:

- El juego va a manejar un menú de opciones que serán las siguientes.
  - Crear Tablero (T). Esta opción permitirá que el programa cree aleatoriamente el tablero del pupiletra, para ello debe elegir un nivel (ver siguiente punto)
  - Jugar (J). Esta opción permitirá simular el juego del pupiletra, mayor detalle en el punto 3. Debe considerar que, para jugar, el jugador debió crear antes su tablero de pupiletras. Si no lo hizo así debe enviarle un mensaje de error y no realizar nada del juego.
  - Salir (S). Esta opción permitirá al jugador terminar y salir del juego.

Si ingresa una opción distinta a las indicadas, debe volver a solicitar el ingreso de la opción.

- El juego va a manejar los siguientes niveles para la creación del tablero:
  - **Experto (X)**. Se debe crear un tablero de pupiletras con letras aleatorias de 30x30.
  - **Personalizado (P)**. Se debe solicitar al usuario que ingrese el número de filas y columnas que tendrá el tablero de pupiletras y se debe generar con letras aleatorias. Considere que el tamaño máximo del tablero será de 40x40, si ingresa un número de filas y/o columnas incorrecto debe solicitar que vuelva a ingresar ambos valores (filas y columnas), hasta que los ingrese correctamente.
- Para jugar, el usuario debe ingresar una cantidad de palabras que va a buscar en el pupiletra. Posterior a ello, debe ingresar cada palabra a buscar y el programa debe verificar si la palabra ingresada se encuentra en el pupiletra. Si la palabra existe dentro del pupiletras debe indicar en qué posición (fila,columna) inicia y en qué posición (fila,columna) termina. Considere que para buscar una palabra solo lo puede hacer de forma vertical (arriba-abajo o abajo-arriba) y horizontal (izquierda-derecha o derecha-izquierda). Si la palabra no existe dentro del pupiletra debe enviar un mensaje indicado lo siguiente: “La palabra buscada no existe en el pupiletra”. Este proceso se va a repetir hasta completar el número de palabras indicado al inicio del juego.
- Si al término de todas las búsquedas, el programa encontró más del 50% de las palabras ingresadas, se considerará que el jugador GANÓ, caso contrario que PERDIÓ.

Se le pide elaborar un programa en C++ que permita a una persona jugar el “PUPI-LPOO” de acuerdo a las indicaciones arriba mencionadas.

Un ejemplo de ejecución del juego sería el siguiente

Bienvenido a PUPU-LPOO.

Menú de opciones:

Crear Tablero (T)

Jugar (J)

Salir (S)

Ingrese su opción: T

Elija el nivel del tablero (X-Experto y P-Personalizado): P

Ingrese el número de filas: 10

Ingrese el número de columnas: 10

A continuación, se muestra el tablero generado.

P	X	T	D	K	F	Z	A	V	C
A	R	B	A	L	A	P	K	A	O
T	S	E	D	S	V	X	F	M	N
O	Z	I	I	H	X	K	N	O	F
L	A	P	R	E	C	I	A	R	I
E	O	Q	U	N	B	U	Z	C	A
R	C	L	G	M	O	E	X	A	R
A	Z	M	E	J	E	S	S	J	P
R	Z	U	S	T	K	H	P	O	Z
M	O	G	O	L	A	I	D	Z	S

Menú de opciones:

Crear Tablero (T)

Jugar (J)

Salir (S)

Ingrese su opción: J

Ingrese la cantidad de palabras a buscar: 3

Ingrese la palabra 1: DIALOGO

La palabra se encuentra en el tablero, inicia en la posición (10,8) y termina en la posición (10,2).

Ingrese la palabra 2: FAMILIA

La palabra no se encuentra en el tablero.

Ingrese la palabra 3: PATO

La palabra se encuentra en el tablero, inicia en la posición (1,1) y termina en la posición (4,1).

El jugador gana porque ha encontrado el 66.67% de las palabras.

Esto se repite en todo el juego hasta el final del mismo.

**NOTA:** Solo puede utilizar las función srand() y time() para generar letras aleatorios y funciones creadas por Ud. **Debe realizar todas las validaciones necesarias para el juego.**

#### Criterios de calificación.

- Por generar correctamente el tablero aleatorio de letras.
- Por imprimir correctamente el tablero aleatorio de letras.
- Por leer correctamente la palabra a buscar.
- Por implementar la lógica de la búsqueda de una palabra en el pupiletras.
- Por implementar la lógica restante en la función main que permita utilizar los puntos anteriores y alguna lógica adicional a las mencionadas.

Lima, 21 de agosto de 2024