

Universidad de San Carlos de Guatemala

Escuela de Ciencias y Sistemas

Facultad de Ingeniería

Introducción a la programación y Computación 1

Primer Semestre 2025

Catedrático:

Tutor académico:



FIUSAC
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

PRÁCTICA 1

Sopa de Letras

Objetivos

Generales

- Familiarizar al estudiante con el lenguaje de programación Java.
- El estudiante aplique los conocimientos adquiridos en el curso de Introducción a la programación y computación 1.
- Elaborar la lógica para presentar una solución a las problemáticas planteadas.

Específicos

- Utilizar el lenguaje de programación Java como herramienta de desarrollo de software
- Construcción de aplicaciones simples en consola.
- Implementación de sentencias de control, ciclos y vectores.
- Aplicación de conceptos de diagramas de flujo.

Desarrollo de la Práctica

La práctica consiste en el desarrollo de una aplicación de consola que simulará un juego de sopa de letras en donde el usuario tendrá que buscar en el tablero las palabras ingresadas en el juego. La interacción del juego con el usuario será de manera sencilla mediante un menú en consola.

Aplicación

Se debe contar con las siguientes funcionalidades:

Nueva Partida: Al escoger una nueva partida, primero se le pedirá el nombre del usuario. Después el usuario podrá elegir entre 3 opciones:

- **Menú Palabras:** En esta opción el usuario podrá insertar, modificar, eliminar o mostrar las palabras que se hayan ingresado.
 - **Insertar:** Se le solicitará al usuario que ingrese el número de palabras que desea ingresar y, posteriormente, deberá introducir esa cantidad de palabras. Si una palabra no cumple con la longitud permitida, se mostrará un mensaje de error y se solicitará nuevamente. Los detalles sobre la longitud de las palabras y el tamaño del tablero se especifican en la [Tabla 1](#).
 - **Modificar:** Deberá ingresar la palabra a modificar y la nueva palabra. Si la nueva palabra no cumple con la longitud permitida, se mostrará un mensaje de error y se solicitará nuevamente. Si la palabra a modificar no se encuentra en el banco de palabras, se mostrará un mensaje de error y se regresará al menú de palabras. Los detalles sobre la longitud de las palabras se especifican en la [Tabla 1](#).
 - **Eliminar:** Deberá de ingresar la palabra que se quiera eliminar y si se encuentra será eliminada del banco de palabras de lo contrario deberá de mostrar un mensaje de error y regresar al menú palabras.
 - **Salir:** Regresará al menú nueva partida.

Tabla 1: Longitud de palabras y tamaño del tablero.

SECCIÓN	LONGITUD DE PALABRAS	TAMAÑO DE TABLERO
A	De 4 a 10 caracteres	17 x 17
B	De 3 a 8 caracteres	15 x 15
C	De 5 a 12 caracteres	20 x 20
D	De 6 a 15 caracteres	25 x 25

E	De 5 a 10 caracteres	14 x 14
F	De 4 a 12 caracteres	18 x 18
G	De 6 a 10 caracteres	16 x 16

- **Jugar:** El usuario podrá comenzar a jugar una partida con las palabras ingresadas. Si las palabras no han sido agregadas deberá de mostrarse un mensaje de error indicando que las palabras no han sido agregadas y mostrar el menú de nuevo el menú de nueva partida.
- **Terminar Partida:** Al escoger esta opción, la partida termina y se regresará al menú principal.

Historial de partidas: En esta opción se deberá de mostrar en consola en nombre del jugador y los puntos obtenidos, número de fallos, cantidad de palabras encontradas de todas las partidas jugadas.

Mostrar puntuaciones más altas: La aplicación deberá contar con una opción para mostrar los jugadores que han finalizado la partida con mayor cantidad de puntos, deberá desplegar en consola de mayor a menor los 3 jugadores con su respectivo nombre y puntuación obtenida.

Mostrar información de estudiante: Se deberá de Imprimir en consola el nombre completo del estudiante, carnet y sección a la que pertenecen.

Salir: Finalizará la ejecución de la aplicación.

Funciones de la Aplicación

Jugar:

Al momento que ya fueron elegidas las palabras ya se podrá empezar a jugar. Se deberán de colocar las palabras ingresadas de forma horizontal y vertical de forma aleatoria en el tablero del tamaño ingresado por el usuario.

Se va a verificar que si se ingresan de nuevo las mismas palabras en otra partida no se encuentren en la misma posición. Las casillas en donde no se encuentren las palabras deberán de ser rellenadas con letras de la A-Z de forma aleatoria. Se debe de mostrar el tablero en consola de forma clara y entendible para el usuario.

Una vez se imprima el tablero, se le solicitará al usuario que ingrese una palabra. Si la palabra que ingreso se encuentra en el banco de palabras, deberá de sustituir la palabra encontrada con los símbolos '#'(Más adelante se puede encontrar un ejemplo de esto) y se le sumará los puntos al jugador.

Si la palabra no se encuentra en el banco de palabras se le quitara una oportunidad. El juego termina cuando el jugador encuentre todas las palabras o se equivoque 4 veces.

Durante el juego, se deberá de mostrar un contador de palabras identificadas y un contador de palabras pendientes a identificar.

Puntos en la partida:

Se deberá de llevar un registro de los usuarios que han jugado y la cantidad de puntos que han acumulado. En cada partida el usuario empezará con 25 puntos. La cantidad de puntos que se le dará por palabra dependerá de la longitud de la misma, Por ejemplo, si una palabra tiene una longitud de 5 ganará 5 puntos. Por cada error se le restará 5 puntos y solo podrá equivocarse 4 veces en una partida.

EJEMPLO:

- ❖ Ejemplo de el ingreso de las palabras para jugar una partida

```
INGRESE 5 PALABRAS ENTRE 4 Y 10 CARACTERES
1) MONITOR
2) TECLADO
3) PANTALLA
4) MOUSE
5) VIRTUAL
Elija una de las siguientes opciones:
1.Menu Palabras
2.Jugar
3.Terminar
```

- ❖ Ejemplo de las palabras en el tablero.

```
|c|u|q|j|t|m|r|g|g|o|v|y|x|y|p|
|w|n|x|u|y|o|d|g|e|e|p|o|v|p|j|
|h|l|b|h|h|w|o|i|s|l|w|c|p|v|i|
|x|l|T|v|j|g|n|i|n|f|l|j|c|k|v|
|j|c|E|c|u|u|z|q|p|q|s|v|t|c|s|
|t|i|C|h|d|w|P|A|N|T|A|L|L|A|r|
|d|i|L|j|h|m|M|O|N|I|T|O|R|o|i|
|p|M|A|a|j|g|x|y|n|o|j|e|f|f|i|
|x|O|D|V|I|R|T|U|A|L|n|q|p|d|j|
|s|U|O|n|y|o|y|d|p|i|n|c|h|x|w|
|m|S|u|m|u|g|q|t|a|b|p|f|n|q|d|
|b|E|z|n|c|n|x|l|u|p|e|i|x|s|j|
```

- ❖ Ejemplo de la palabra PANTALLA encontrada en el tablero:

	c		u		q		j		t		m		r		g		g		o		v		y		x		y		p	
	w		n		x		u		y		o		d		g		e		e		p		o		v		p		j	
	h		l		b		h		h		w		o		i		s		l		w		c		p		v		i	
	x		l		T		v		j		g		n		i		n		f		l		j		c		k		v	
	j		c		E		c		u		u		z		q		p		q		s		v		t		c		s	
	t		i		C		h		d		w		#		#		#		#		#		#		#		#		#	
	d		i		L		j		h		m		M		O		N		I		T		O		R		o		i	
	p		M		A		a		j		g		x		y		n		o		j		e		f		f		i	
	x		O		D		V		I		R		T		U		A		L		n		q		p		d		j	
	s		U		O		n		y		o		y		d		p		i		n		c		h		x		w	
	m		S		u		m		u		g		q		t		a		b		p		f		n		q		d	
	b		E		z		n		c		n		x		l		u		p		e		i		x		s		j	

Requerimientos

Documentación

- Manual técnico (descripción de los métodos creados y requerimientos de la aplicación) en PDF.
- Manual de usuario (Cómo funciona la aplicación y cómo el usuario interactúa con ella) en PDF.
- Diagrama de flujo del proceso general en PDF.

Consideraciones

- El tablero de la sopa de letras debe de ser una matriz según como lo indica la Tabla 1.
- Las palabras que se mostrarán en el tablero serán ingresadas por el usuario.
- Las palabras solo podrán ser ingresadas de manera horizontal y vertical.
- La partida termina cuando el usuario se equivoque más de 4 veces.
- Se debe de validar que las palabras tengan la longitud indicada en la Tabla 1.
- El programa continuará con la ejecución hasta que se seleccione la opción salir.
- La interacción del usuario con la aplicación será mediante la lectura del teclado.
- En todo el programa no se distinguirá entre mayúsculas y minúsculas.
- Se podrá jugar N cantidad de partidas sin que termine el programa y sin que se pierdan las estadísticas.

Restricciones

- La aplicación debe ser desarrollada en el lenguaje de programación JAVA.
- No se permite utilizar código copiado o bajado de internet.
- El IDE por utilizar queda a discreción del estudiante (se recomienda el uso de NetBeans).
- Las copias obtendrán nota de 0 y reporte a la Escuela de Ciencias y Sistemas.
- Durante la calificación se le solicitará al estudiante modificar el código del proyecto con el objetivo de validar la creación de este.
- El estudiante no tendrá derecho a calificación si no muestra sus dos hojas de calificación impresas en el día de la práctica.
- Tiene que contar con mínimo 2 commits semanales y tener agregado al auxiliar de colaborador del repositorio para monitorear los avances realizados en el proyecto, de lo contrario obtendrá una penalización.

Habilidades por evaluar

- Uso de variables globales y locales.
- Uso de memoria estática.
- Uso de estructuras de control y de selección.
- Uso correcto de los arreglos y matrices.
- Conocimientos sobre sistemas computacionales.
- Habilidad para analizar y sintetizar información.
- La habilidad de comprender y realizar diagramas de flujo.
- Habilidad para resolver problemas.

Entrega

- **FECHA DE ENTREGA:** 14/02/2025 antes de las 23:59 (No se aceptarán entregas, ni commits a partir de esa fecha y hora).
- En su repositorio adjuntar código fuente y la documentación solicitada.
- El repositorio debe ser privado y en Github, teniendo el nombre de *IPC1_Practica1_carnet*, por ejemplo: *IPC1_Practica1_202400000*.
- Subir el enlace del repositorio de Github en la tarea asignada en UEDI.
- Agregar de colaborador del repositorio al auxiliar correspondiente a su sección antes del 14/02/2025, de lo contrario habrá penalización:

Sección	Auxiliar	Usuario de github
A	Anthony Alexander Aquino Santiago	anthony-97
B	Sebastian Alejandro Velasquez Bonilla	SebastianVelasquez22901
C	Juan Francisco Urbina Silva	Iskandar1412
D	Lesther Kevin Federico López Miculax	LestherLopez
E	Douglas Alexander Soch Catalán	DAlexSC
F	Zenaida Irazema Chacón García	ZenaidaChacon
G	Max Rodrigo Durán Canteo	