**Instituto Nacional de Aprendizaje**

**Programador de Aplicaciones Informáticas**

**Módulo Lógica Computacional**

**Primera Fase del Proyecto Final**

**Estudiante: Mariel Daniela Rojas Sanchez**

**Cedula: 208030487**

**Profesor: Oscar Pacheco Vásquez**

**20 Julio 2022**

Índice

[Introducción: 3](#_Toc111751875)

[Objetivo General: 4](#_Toc111751876)

[Objetivos Específicos: 4](#_Toc111751877)

[Problemas 4](#_Toc111751878)

[EJERCICIO 1: Juego Educativo Memoria 4](#_Toc111751879)

[EJERCICIO 2: Cifrado de Texto - Playfair 5](#_Toc111751880)

[Juego Educativo Memoria 7](#_Toc111751881)

[Planteamiento del Problema: 7](#_Toc111751882)

[Identificación del Problema: 7](#_Toc111751883)

[Datos de entrada: 7](#_Toc111751884)

[Datos de salida: 7](#_Toc111751885)

[Observaciones (Limitaciones / Excepciones / Suposiciones): 7](#_Toc111751886)

[Planteamiento de alternativas: 8](#_Toc111751887)

[Alternativa 1: 8](#_Toc111751888)

[Alternativa 2: 9](#_Toc111751889)

[Elección de alternativa: 11](#_Toc111751890)

[Cifrado de Texto – Playfair 12](#_Toc111751891)

[Planteamiento del Problema: 12](#_Toc111751892)

[Identificación del Problema: 12](#_Toc111751893)

[Datos de entrada: 12](#_Toc111751894)

[Datos de salida: 12](#_Toc111751895)

[Observaciones (Limitaciones / Excepciones / Suposiciones): 12](#_Toc111751896)

[Planteamiento de alternativas: 13](#_Toc111751897)

[Alternativa 1: 13](#_Toc111751898)

[Alternativa 2: 15](#_Toc111751899)

[Elección de alternativa: 16](#_Toc111751900)

[Conclusiones: 16](#_Toc111751901)

[Bibliografía: 17](#_Toc111751902)

# Introducción:

Este proyecto tiene como fin poner en practica todos los temas vistos a lo largo del módulo de lógica computacional; En este módulo se aprende sobre el razonamiento lógico, la metodología de soluciones de problemas y por último a la realización de algoritmos y diagramas de flujo, según distintas solicitaciones.

El uso de los conocimientos aprendidos a lo largo del módulo es de suma importancia, para tener una buena base en la formación como estudiantes, fomentando distintos temas y cualidades características de un buen programador, como por ejemplo aprender a aplicar la lógico-matemática, abstracción e interpretación de distintos problemas, lógica proposicional y por último a la lógica computacional.

En este proyecto se van a analizar dos ejercicios distintos, en ambos ejercicios se va a ejecutar un planteamiento de problema, se va a identificar el problema estableciendo sus datos de entrada, salida, proceso, así como también se toman en cuenta diferentes observaciones, validaciones y limitaciones, seguidamente se hace un planteamiento de dos distintas alternativas, para la elección de la alternativa más óptima, por último se diseña un algoritmo que aporte una solución adecuada a los ejercicios planteados.

# Objetivo General:

Resolver problemas empresariales mediante la metodología de solución de problemas, y así mismo crear algoritmos, diagramas de flujo óptimos según una serie de requerimientos presentados.

# Objetivos Específicos:

Determinar las posibles soluciones a el problema del juego de memoria y del cifrado de playfair, mediante la resolución de la plantilla de método de solución de problemas.

Implementar un pseudocódigo optimo, que sea capaz de jugar el juego de memoria y otro que también que resuelva el cifrado de playfair.

Diseñar diagramas de flujo funcionales y entendibles para cada uno de los problemas planteados.

# Problemas

## EJERCICIO 1: Juego Educativo Memoria

Este es un juego que se desarrolla con pares de cartas que tienen un mismo símbolo. Las cartas se colocan sobre una mesa, con las figuras hacia abajo, se mezclan, y posteriormente, cada jugador toma un turno para dar vuelta a dos cartas (cuáles quiera), si las cartas que giró tienen la misma figura entonces se le suma un punto y tiene la oportunidad de virar otras dos cartas. Cuando un jugador vira dos cartas y las mismas no comparten la misma figura, entonces las cartas se “tapan” nuevamente y se dejan en la misma posición en la que estaban, el jugador pierde su turno y continua el siguiente. El juego se termina cuando ya no quedan cartas por volver. El jugador que haya acumulado más puntos es quien gana la partida.

En el juego se pueden presentar empates, lo cual sucede cuando varios jugadores alcanzaron el máximo puntaje.

Para este ejercicio solo habrá 2 jugadores, uno debe ser el usuario humano y el otro jugador será la computadora. Al preparar este algoritmo se debe lograr que la computadora juegue de manera inteligente, y que no solo realice jugadas aleatorias.

Se requiere que usted diseñe un algoritmo que:

✓ Le permita a un jugador humano, jugar el juego de memoria contra una computadora(inteligente).

✓ Sea capaz de determinar si hay algún ganador o si hay empate.

✓ Controle que se alterne el jugador que inicia en cada turno. El primer turno de cada ronda será aleatorio (Puede iniciar cualquiera de los 2, pero los siguientes se deben alternar)

✓ Permita seleccionar el nivel de dificultad entre 5 parejas, 9 parejas y 13 Parejas.

## EJERCICIO 2: Cifrado de Texto - Playfair

Desde que se inventó la escritura se ha utilizado el texto para pasar mensajes desde un lugar a otro, o más bien, desde un emisor hasta un receptor. Y ha habido momentos de la historia en los que el mensaje que se quiere enviar es tan importante que se quiere proteger o mantener en secreto, para que las personas no autorizadas no puedan recibir información que no deberían conocer. Ya que esa necesidad de cifrar un mensaje es casi tan antigua como la escritura misma, a lo largo del tiempo se han desarrollado distintos métodos para cifrar y descifrar mensajes de texto.

El cifrado de Playfair fue el primer sistema de cifrado en encriptar pares de letras. Wheatstone inventó el cifrado para encriptar mensajes enviados por telegrama, pero lleva el nombre de su amigo lord Playfair, quien lo promovió para uso militar. Este método fue considera “irrompible” por mucho tiempo, tanto así que la Fuerzas Armada de Inglaterra y USA utilizaron este sistema para encriptar sus comunicaciones durante las Guerras Mundiales.

Este sistema utilizaba una llave simétrica, lo que significa que la misma clave se usa tanto en el Cifrado como en el Descifrado. Este sistema consiste en separar el texto “original” en pares de letras y proceder a su cifrado de acuerdo con una matriz alfabética de dimensiones 5x5 en la cual se encuentran representadas las 26 letras del alfabeto inglés, tomando en cuenta que la I y la J comparten el mismo espacio de la matriz. Para que este método de cifrado presente un mayor nivel de seguridad, se incluirá al comienzo de

dicha matriz una clave que se escribe a partir de la primera fila omitiendo las letras repetidas. A continuación de dicha clave, se distribuyen las restantes letras del alfabeto hasta completar toda la matriz, en la matriz NO deben aparecer letras repetidas. Este método de cifrado debe cumplir las siguientes condiciones:

1. Si los caracteres M1 y M2 están en la misma fila, C1 y C2 son los dos caracteres de la derecha.

2. Si M1 y M2 están en la misma columna, C1 y C2 son los dos caracteres de abajo.

3. Si M1 y M2 están en filas y columnas distintas, C1 será la letra de la intersección horizontal del M1 respecto al M2 y C2 será la letra de la intersección vertical del M1 respecto al M2.

4. Si M1 = M2, insertar carácter sin significado entre m1 y m2 para evitar su repetición, y después aplicar las reglas 1-3.

5. Si el número de letras, del mensaje original, es impar entonces se debe añadir una letra sin significado al final del texto.

**Notas**:

✓ M1 y M2 son los caracteres de los pares del mensaje Original.

✓ C1 y C2 representan los caracteres del correspondiente par del mensaje cifrado.

✓ El conteo de los caracteres en la matriz es circular.

Las condiciones dadas explican las reglas para lograr Cifrar un mensaje. De manera lógica podemos inferir que el Descifrado se aplica invirtiendo las reglas dadas.

Ahora bien, usted ha sido contratado como desarrollador en la CCSS para prevenir el robo de información del grupo de hackers llamado CONTI. Para esto se requiere que usted desarrolle un algoritmo que sea capaz de Cifrar y Descifrar texto utilizando el método Playfair, tome en cuenta lo siguiente:

✓ Debe ser capaz de cifrar un mensaje utilizando una clave(llave) indicada por el usuario.

✓ Debe poder descifrar un mensaje encriptado, si se ingresa la clave correcta.

# Juego Educativo Memoria

## Planteamiento del Problema:

Desarrollar un juego de memoria, el cual consta de parejas de cartas y tiene varios grados de dificultad, donde solo puede haber un ganador.

## Identificación del Problema:

### Datos de entrada:

* Nombre del jugador.
* Nivel de dificultad deseado.
* Número para empezar el juego.
* Posición de las cartas por seleccionar.

### Datos de salida:

* Mensaje imprimiendo el nombre y la dificultad seleccionada.
* Mensaje solicitando un número para iniciar el juego.
* Interfaz para visualizar las cartas.
* Mensaje imprimiendo el resultado final.

## Observaciones (Limitaciones / Excepciones / Suposiciones):

* El juego se termina cuando ya no quedan cartas por volver, sin embargo, el usuario puede volver a jugar o finalizar por completo.
* El jugador que haya acumulado más puntos es quien gana la partida.
* En el juego no se pueden presentar empates.
* El primer turno de cada ronda será aleatorio, pero luego del primer turno los siguientes se deben alternar entre ambos jugadores.
* Existen 3 niveles de dificultad, entre 5 parejas (Matriz 5x2), 9 parejas (Matriz 9x2) y 13 Parejas (Matriz 13x2).
* Únicamente habrá 2 jugadores, el usuario y la computadora.
* La computadora jugara inteligentemente.
* Cada jugador toma un turno para dar vuelta a dos cartas (cuáles quiera), si las cartas que giró tienen la misma figura entonces se le suma un punto y tiene la oportunidad de virar otras dos cartas.
* Cuando un jugador vira dos cartas y las mismas no comparten la misma figura, entonces las cartas se “tapan” nuevamente y se dejan en la misma posición en la que estaban, el jugador pierde su turno y continua el siguiente.

## Planteamiento de alternativas:

### Alternativa 1:

1. Inicio.
2. Se define todas las variables necesarias.
3. Se le da la bienvenida al usuario.
4. Solicitar nombre del jugador.
5. Lee el nombre.
6. Nombre es igual a jugador usuario.
7. Escribir en pantalla que el juego tiene 3 niveles de dificultad, de 5,9 y 13 parejas.
8. Leer nivel.
9. Según el nivel digitado se le imprime al usuario el nivel escogido.
10. Si nivel es igual a 5 entonces la dimMatriz es de 5x2, y se llena con valores del 1 al 5, los cuales se repiten dos veces para formar pares, Si no, si nivel es igual a 9 entonces la dimMatriz es de 9x2, y se llena con valores del 1 al 9, los cuales se repiten dos veces, si no, si nivel es igual a 13 entonces la dimMatriz es de 13x2, y se llena con valores del 1 al 13, los cuales se repiten dos veces.
11. Se debe realizar un recorrido para comprobar que los valores no se repitan más de dos veces.
12. Cuando la matriz este definida y completa, se imprime en pantalla la matriz mostrando únicamente las posiciones fila y columna de la matriz.
13. Se hace aleatoriamente para ver cual jugador inicia.
14. Se realiza un ciclo repetir hasta que no existan posiciones en la matriz por voltear.
15. Se le solicita al jugador escoger la posición de fila y columna de dos cartas a voltear.
16. Se leen las posiciones de fila y columna ingresadas por el usuario o escogidas inteligentemente por la computadora.
17. Se imprime en pantalla los dos dígitos que contenían ambas posiciones, si los dígitos son iguales, se le suma un punto a ese jugador ya sea jugador compu o jugador usuario, dicho jugador que adivino puede volver a elegir cartas, Si no, no se le suma punto y continua el otro jugador.
18. Se imprime en pantalla que jugador compu obtuvo “X puntos” y que jugador usuario obtuvo “X puntos”.
19. Si puntos de jugador usuario son más que el de jugador compu, imprimir que gano, si no, imprimir que perdió.
20. Fin.

### Alternativa 2:

1. Inicio.
2. Se define todas las variables necesarias.
3. Se le da la bienvenida al usuario.
4. Solicitar nombre del jugador.
5. Lee el nombre.
6. Escribir en pantalla que el juego tiene 3 niveles de dificultad, de 5,9 y 13 parejas y se le solicita que digite un nivel.
7. Leer nivel.
8. Según el nivel digitado se le imprime al usuario el nivel escogido, si el usuario digito 1 entonces serian 5 cartas, si digito 2 serian 9 cartas, digito 3 serian 13 cartas.
9. Se le solicita al usuario que digite un numero para inicial el juego.
10. Se dimensionan los vectores a utilizar, en este caso existe un vector que contiene las cartas con el nombre de distintos países, otro vector que contiene las posiciones a elegir, otro vector que contiene la cantidad de cartas que el usuario selecciono, un vector que contiene todo el mazo completo, que sería la cantidad de cartas multiplicadas por dos, y por último un vector que es el encargado de revolver las cartas.
11. El vector cartas se iguala a los nombres de los países.
12. El vector de posiciones se iguala a las posiciones, pero en este caso de 1 a 13, evitando el cero.
13. La cantidad de cartas elegidas se iguala al vector que contiene las posiciones, esto se realiza con un ciclo para que vaya recorriendo todos los espacios del vector.
14. Se llena el vector de mazo completo con los valores de las cartas, dependiendo de la cantidad de cartas selecciones.
15. Se realiza un ciclo repetitivo donde se va a verificar que no existan valores repetidos en el mazo que va a ser revuelto.
16. Se imprime en pantalla que seleccione una posición del vector para destapar las cartas.
17. Se imprime en la pantalla el vector mostrando únicamente cantidad de cartas \*2 del nivel escogido.
18. Se hace aleatoriamente para ver cual jugador inicia.
19. En un ciclo repetitivo hasta que se no hayan cartas por voltear se hace lo siguiente:
20. Se le solicita al jurador que fue seleccionado mediante la función aleatorio, que “x” escoja en las posiciones del vector de dos cartas a voltear.
21. Se leen las posiciones escogidas de cada carta escogida por el usuario o por la computadora, si las cartas seleccionadas fueron iguales entonces se le suma un punto a ese jugador ya sea jugador compu o jugador y dicho jugador que adivino puede volver a elegir cartas, Si no, no se le suma punto y continua el siguiente jugador.
22. Para la elección de cartas en el caso de la computadora, la computadora va a elegir dos cartas al azar, pero lo va a hacer inteligentemente, por lo que existirá una variable que va a guardar todas las cartas y sus respectivas posiciones para que en la próxima jugada pueda elegir inteligentemente las cartas que guardo en la variable y que son iguales, si no hay cartas abiertas que sean iguales entonces la compu va a elegir una carta que ya fue abierta pero no emparejada con una que no ha sido abierta para intentar de hacer pareja.
23. Cada jugada se imprime en pantalla la cantidad de puntos que lleva cada jugador.
24. Si puntos de jugador usuario son más que el de jugador compu, imprimir que gano, si no, imprimir que perdió.
25. Fin.

## Elección de alternativa:

Considero que la alternativa 2 está mucho mejor desarrollada, es más sencilla y optima de aplicar como programadores, ya que al utilizar vectores estos únicamente utilizan una fila, ya que son unidimensionales, por lo que se hace más sencillo de trabajar, en cuanto a lo visual o lo eficaz va a depender de la cantidad de funciones o procedimientos que se utilicen, sin embargo, un vector necesita menos espacio en memoria y es más fácil de recorrerlo.

# Cifrado de Texto – Playfair

## Planteamiento del Problema:

Desarrollar un algoritmo que sea capaz de Cifrar y Descifrar texto utilizando el método Playfair,

## Identificación del Problema:

### Datos de entrada:

* Clave ingresada por el usuario.
* Mensaje que desea descifrar o cifrar.
* Opción de cifrar o descifrar.

### Datos de salida:

* Matriz llenada correctamente con la clave.
* Mensaje cifrado o descifrado.
* Menú preguntando si se desea cifrar o descifrar.

## Observaciones (Limitaciones / Excepciones / Suposiciones):

* La matriz es de 5x5.
* La matriz contiene únicamente letras alfabéticas (26 letras).
* La I y la J comparten el mismo espacio de la matriz, únicamente estas letras pueden compartir la misma posición en la matriz.
* En la matriz no deben aparecer letras repetidas.
* Para el descifrado se usan las reglas contrarias del cifrado.
* Si los caracteres M1 y M2 están en la misma fila, C1 y C2 son los dos caracteres de la derecha.
* Si M1 y M2 están en la misma columna, C1 y C2 son los dos caracteres de abajo.
* Si M1 y M2 están en filas y columnas distintas, C1 será la letra de la intersección horizontal del M1 respecto al M2 y C2 será la letra de la intersección vertical del M1 respecto al M2.
* Si M1 = M2, insertar carácter sin significado entre M1 y M2 para evitar su repetición, y después aplicar las reglas 1-3.
* Si el número de letras, del mensaje original, es impar entonces se debe añadir una X sin significado al final del texto.
* M1 y M2 son los caracteres de los pares del mensaje Original.
* C1 y C2 representan los caracteres del correspondiente par del mensaje cifrado.
* El conteo de los caracteres en la matriz es circular.
* En un caso el usuario puede digitar la clave, en otros la clave puede estar predefinida.
* El algoritmo debe ser capaz de cifrar un mensaje utilizando una clave (llave) indicada por el usuario y además debe poder descifrar un mensaje encriptado, si se ingresa la clave correcta.

## Planteamiento de alternativas:

### Alternativa 1:

1. Inicio.
2. Se definen todas las variables a utilizar.
3. Se dimensiona la matriz que es de 5x5.
4. Se crea un menú donde según el número que el usuario digito puede elegir si desea descifrar, cifrar o bien salir del código.
5. Se crea un subproceso que se llama en el menú, donde se le solicita que digite la clave el mensaje que desea cifrar o descifrar.
6. Se lee la clave y el mensaje.
7. Se crea una función que se va a encargar de eliminar las letras repetidas de la clave ingresada por el usuario.
8. Se crea otra función donde se van a eliminar las letras repetidas de la clave y del abecedario.
9. Se crea un subproceso que es el encargado de llenar correctamente la matriz con la calve y el abecedario que previamente fueron verificadas si no contenían letras repetidas y además se va a imprimir dicha matriz en pantalla, para que el usuario tenga más interacción con el algoritmo.
10. Se crea una función que verifica si el mensaje es impar se le debe agregar una X al final, si contiene una Ñ se debe sustituirla por una N, si digita una J se cambia por una I y si el mensaje tiene espacios en blanco se debe eliminarlos, si no se debe dejar como esta.
11. Se crea una función que utiliza la matriz llena, esta función que tiene como fin, averiguar las posiciones en la matriz de cada una de las letras del mensaje ingresado, esto lo hace tomando letras de dos en dos, para proceder a realizar el cifrado y descifrado, el cual necesita todas las posiciones de todos los pares de letras del mensaje ingresado.
12. Se crean dos subprocesos aparte, que son muy parecidos, pero uno se encarga de cifrar y otro de descifrar.
13. En ambos procesos es necesario utilizar la función que determina las posiciones y si la letra 1 se encuentra en la misma fila que la letra 2, se verifica primero en cual columna esta, si esta en la ultima columna es necesario iniciarla en cero otra vez, si no, entonces ambas letras se mueven hacia la derecha manteniendo la fila, únicamente cambia la columna, este mismo procedimiento se realiza en el descifrado, pero en este caso se hace el movimiento de las nuevas letras hacia las columnas de la izquierda. Si la letra 1 se encuentra en la misma columna que la letra 2, se verifica primero en cual fila esta, si está en la última fila es necesario iniciar la fila en cero otra vez, si no entonces únicamente ambas letras se mueven hacia abajo, este mismo procedimiento se realiza en el descifrado, pero en este caso se hace el movimiento de las nuevas letras es de la fila para arriba. Si no, si la letra1 y 2, están en filas y columnas distintas, el valor nuevo de la columna de la letra 1 es igual al valor de la columna donde estaba la letra 2, el valor nuevo de la fila de la letra 1 es igual al valor de la fila donde estaba la letra 1, el valor nuevo de la columna de la letra 2 es igual al valor de la columna donde estaba la letra 1 y por último el valor nuevo de la fila de la letra 2 es igual al valor de la fila donde estaba la letra 2. El proceso de las diagonales es el mismo en el cifrado y en el descifrado.
14. El mensaje indicando el mensaje cifrado o descifrado se imprime en pantalla.
15. Fin Algoritmo.

### Alternativa 2:

1. Inicio.
2. Se definen las variables necesarias.
3. Se dimensiona la matriz que sea 5x5.
4. Se le solicita al usuario si quiere cifrar o descifrar.
5. Se le solicita al usuario que ingrese el mensaje que quiere descifrar o cifrar.
6. Se le solicita la clave.
7. Verificar el número de letras que contiene el mensaje ingresado, si el mensaje es impar se le debe agregar una X al final, si no se debe dejar como esta.
8. Dependiendo de la cantidad de letras que contiene el mensaje, se le hace una separación de dos en dos letras al mensaje. (Silabas de 2).
9. Se llena la matriz ordenadamente por fila y columna, primero se ingresa letra por letra la clave en cada espacio de fila y columna, después se llena ordenadamente con las letras faltantes del abecedario.
10. Verificar si en la matriz no se repite ninguna letra, si se repite alguna letra, es necesario eliminarla.
11. Se debe hacer una búsqueda por toda la matriz buscando el primer conjunto de silabas del mensaje (m1y m2), si el conjunto se encuentra en la misma fila entonces c1 se encuentra a la derecha de m1 y c2 a la derecha de m2. Si no, si el conjunto m1 y m2 se encuentra en la misma columna, entonces c1 se van a encontrar debajo de m1 y c2 se va a encontrar debajo de m2; Si no, si m1 y m2 se están en filas y columnas distintas, c1 será la letra de la intersección horizontal del m1 respecto al m2 y c2 será la letra de la intersección vertical del m1 respecto al m2. Si lo anterior no se cumple y m1 y m2 se debe insertar un carácter sin significado para que no se repita y se debe continuar. Todo lo anterior se debe repetir por todos los conjuntos de silabas del mensaje, hasta obtener el mensaje cifrado.
12. Si se quiere descifrar el mensaje, el usuario deberá de escribir la clave y se repite el paso 12, solo que con c1 y c2 como las letras proporcionadas del mensaje y m1 y m2 como las letras por descifrar.
13. Cuando se tenga el mensaje cifrado o descifrado, se imprime en la pantalla.
14. Fin del algoritmo.

## Elección de alternativa:

La primera alternativa es más eficaz, optima y más sencilla de implementar, además esta alternativa utiliza programación modular, por lo que cada problema necesario para resolver el cifrado se dividió en partes, a la hora de revisar el seudocódigo para encontrar errores es más sencillo.

# Conclusiones:

En este proyecto final del módulo de lógica computacional se han puesto en práctica básicamente todos los temas estudiados a lo largo del curso, como lo son: el razonamiento lógico, la metodología de soluciones de problemas y por último la realización de algoritmos para resolver los dos ejercicios planteados.

La práctica de los temas vistos en este curso, siendo estos el razonamiento lógico, la metodología de soluciones de problemas y la realización de algoritmos para resolver los dos ejercicios planteados, nos permite tener una excelente base en la formación hacia un buen programador.

Gracias la metodología de soluciones de problemas realizada en ambos ejercicios se puede analizar de una manera lógica y más a fondo dichos ejercicios, para darles una solución factible y optima que beneficia tanto a el programador como al cliente.

# Bibliografía:

Salazar Carlos (2022). LÓGICA DE PROGRAMACIÓN. Quito-Ecuador: CEMLAD.

Esquivel Andrés. (2015). CIFRADO DE PLAYFAIR. de Making Code. Obtenido del sitio web: <https://www.makingcode.dev/2015/07/cifrado-de-playfair.html>

AbhayBhat. (2022). Cifrado de Playfair con ejemplos. Obtenido del sitio web: <https://www.geeksforgeeks.org/playfair-cipher-with-examples/>