

JUGADOR 1



PUNTUACIÓN MÁS ALTA 2500

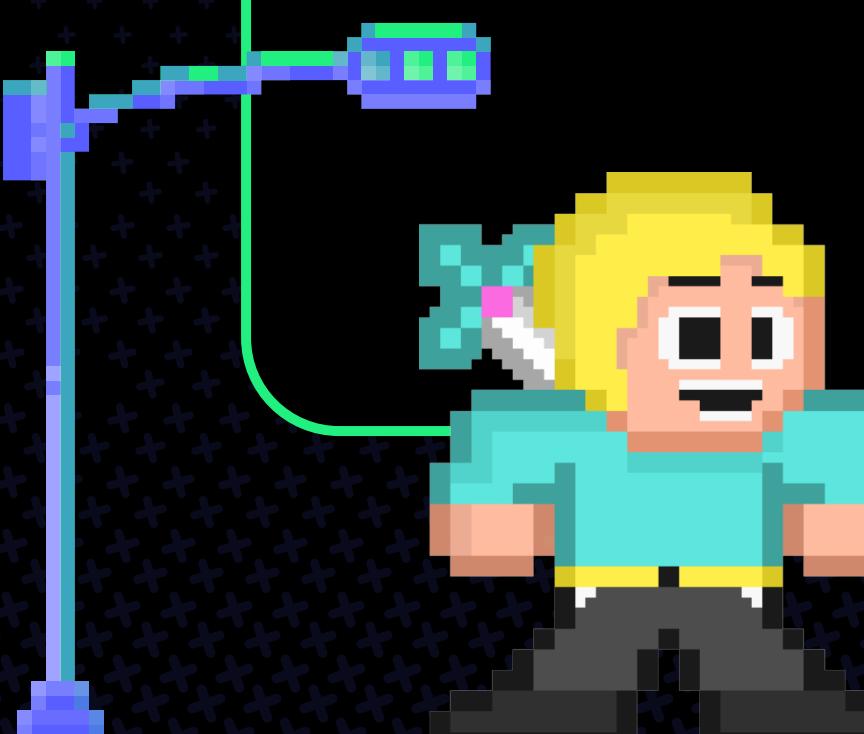


JUGADOR 2

ACTIVIDAD

INTEGRADORA

START



- ◆ MARÍA FERNANDA MORENO GOMÉZ
- ◆ JOSE RICARDO ROSALES CASTAÑEDA
- ◆ URI JARED GOPAR MORALES

MENU

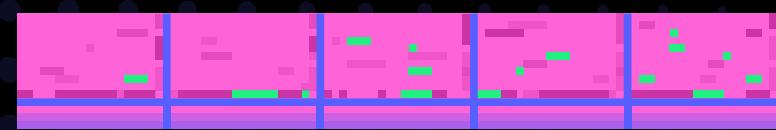
➡ 01

♦ 07

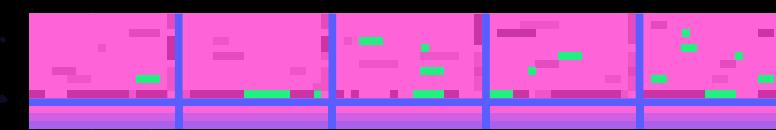
★ 12



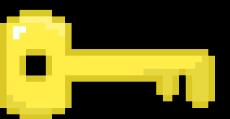
# CONTENIDO



DESCRIPCION



ALGORITMO KMP



ALGORITMO LCS



ALGORITMO  
 MANACHER



# DESCRIPCION

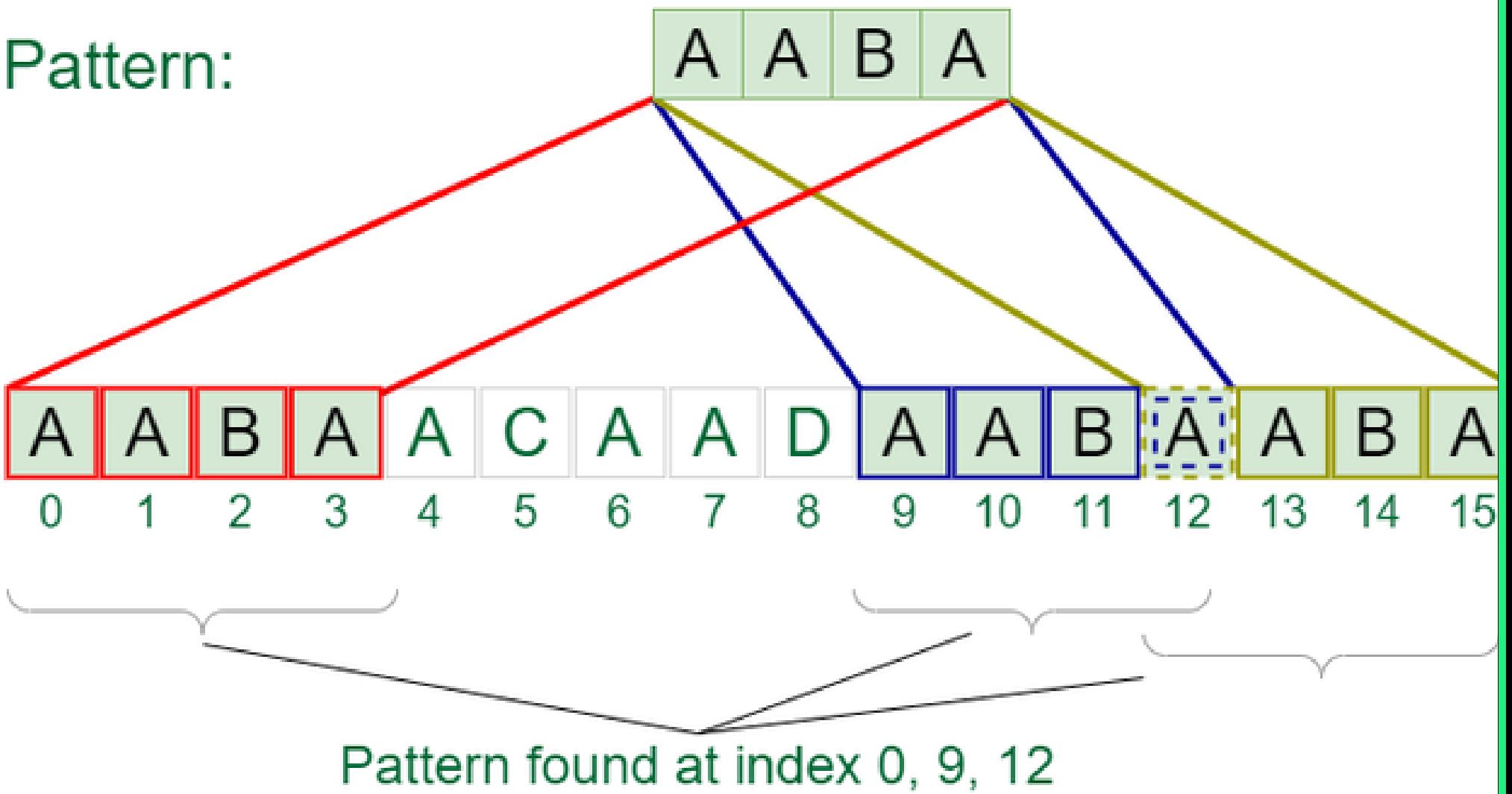
→ DADOS 2 ARCHIVOS DE TRANSMISIÓN (TRANSMISSION1.TXT Y TRANSMISSION2.TXT), LOS CUALES CONTIENEN CARACTERES DE TEXTO REPRESENTANDO EL ENVÍO DE DATOS ENTRE DISPOSITIVOS Y DADOS 3 ARCHIVOS DE CÓDIGO MALICIOSO (MCODE1.TXT, MCODE2.TXT, MCODE3.TXT) QUE SE PUEDEN ENCONTRAR DENTRO DE UNA TRANSMISIÓN, CREAR UN PROGRAMA QUE ANALICE CONCURRENCIAS DE LOS CÓDIGOS MALICIOSOS EN LAS TRANSMISIONES, EL PALÍNDROMO MÁS LARGO EN CADA TRANSMISIÓN Y LA SUBCADENA COMÚN MÁS LARGA EN AMBOS ARCHIVOS.



# ALGORITMO KMP

Text: A A B A A C A A D A A B A A B A

Pattern:



Decidimos implementar el algoritmo de Knuth-Morris-Pratt (KMP) debido a que su complejidad es  $O(n+m)$ . Es más eficiente en cuestión de tiempos cuando se trata de textos extensos y en almacenamiento espacial

Primero debemos de crear la función de LPS(Longest Prefix Suffix), esta tiene como propósito evitar las comparaciones innecesarias, ya que almacena la longitud del prefijo más largo.

Ahora es momento de buscar el texto, para esto inicializamos dos índices los cuales van a ser el texto y el patrón a buscar, para esto si el texto y el patrón coinciden se le agraga uno a la posición, y si se encuentra una ocurrencia del patrón en el texto se almacena en la posición de inicio de la ocurrencia y se ajusta j LPS.

# ALGORITMO LCS

IMPLEMENTAMOS ESTE ALGORITMO PARA PODER ENCONTRAR LA SUBCADENA MÁS LARGA ENTRE DOS O MÁS SECUENCIAS.

PARA SU IMPLEMENTACIÓN CREAMOS UNA MATRIZ  $(M+1) * (N+1)$  DONDE SON LAS LONGITUDES QUE DESEAMOS COMPARAR. PARA ESTO COMPARAMOS LOS ELEMENTOS DE LAS DOS SECUENCIAS EN LAS POSICIONES CORRESPONDIENTES Y SI ESTOS SON IGUALES, EL VALOR EN LA CASILLA ACTUAL DE LA TABLA SE OBTIENE SUMANDO 1 AL VALOR DIAGONAL SUPERIOR Y SI NO ES TOMANDO EL MÁXIMO ENTRE EL VALOR DE LA IZQUIERDA Y EL VALOR DE ARRIBA.

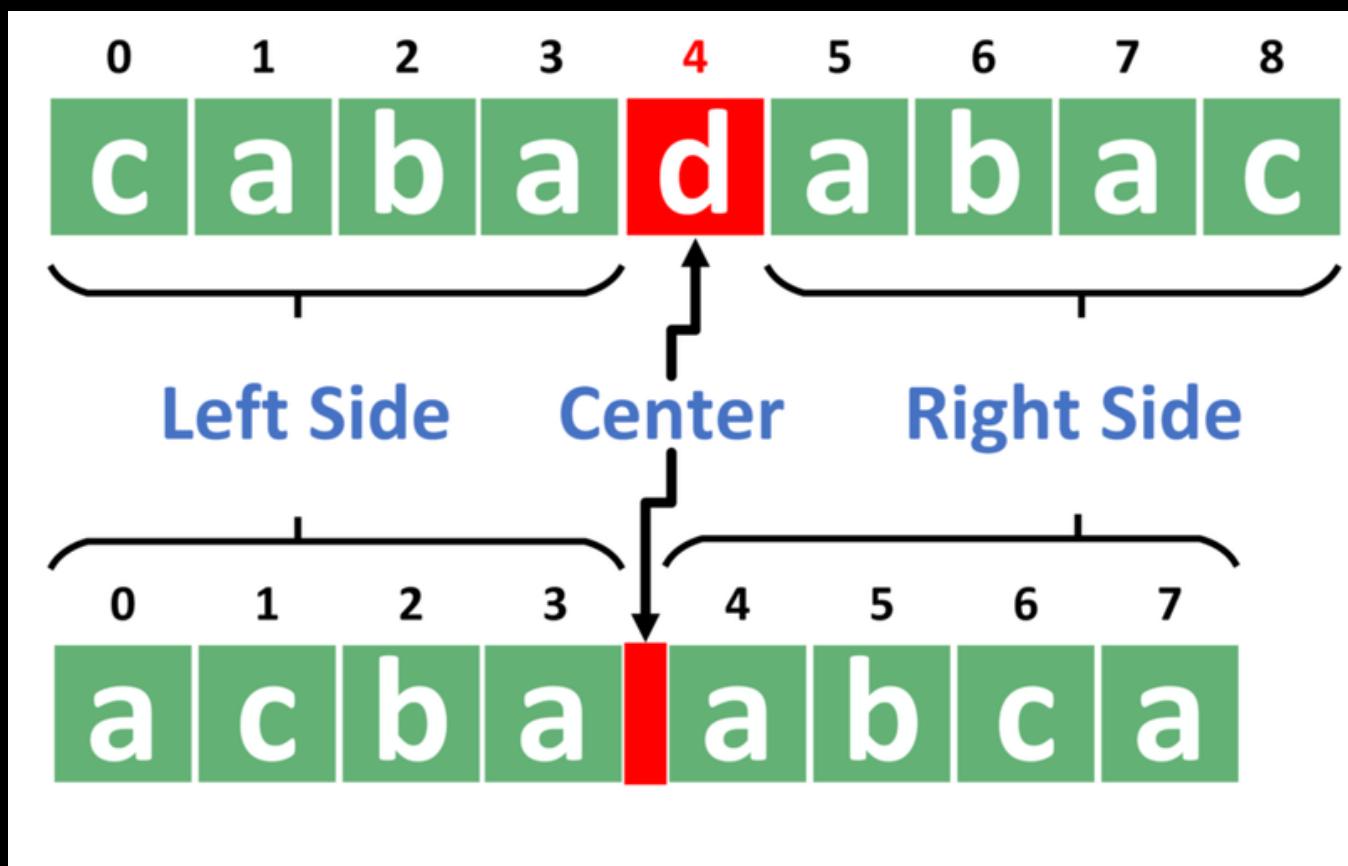
PARA LA SECUENCIA DE X= ABCDGH  
Y=AEDFHR

I	I	I	A	B	C	D	G	H	I
I	I	0	0	0	0	0	0	0	I
I	A	0	1	1	1	1	1	1	I
I	E	0	1	1	1	1	1	1	I
I	D	0	1	1	2	2	2	2	I
I	F	0	1	1	2	2	2	2	I
I	H	0	1	1	2	2	2	3	I
I	R	0	1	1	2	2	2	3	I

MENU



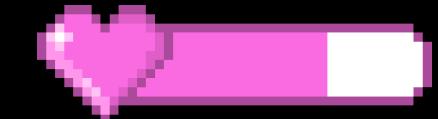
# ALGORITMO MANACHER



LO IMPLEMENTAMOS PARA PODER ENCONTRAR LA SUBCADENA PALÍNDROMA MÁS LARGA, ESTE ALGORITMO A MEDIDA QUE RECORRE LA CADENA, MANTIENE UNA DERECHA QUE REPRESENTA EL BORDE DERECHO DEL PALÍNDROMO MÁS LARGO Y UN CENTRO QUE REPRESENTA LA MITAD DEL PALÍNDROMO.

SE ITERA A TRAVÉS DE CADA CARÁCTER DE LA CADENA, DE ESTE MODO SE CALCULA LA LONGITUD PALÍNDROMA ALREDEDOR DEL CENTRO Y SE ALMACENA EN UN ARRAY. SE UTILIZA EL CENTRO Y LA DERECHA PARA DETERMINAR SI SE PUEDE APROVECHAR LA SIMETRÍA PARA EVITAR COMPARACIONES INNECESARIAS

MENU



MUCHAS  
GRACIAS!