

UNIVERSIDAD NACIONAL DE SAN AGUSTÍN
FACULTAD DE INGENIERIA DE PRODUCCION Y SERVICIOS
DEPARTAMENTO DE INGENIERÍA DE SISTEMAS
LABORATORIO - ANÁLISIS Y DISEÑO DE ALGORITMOS
GABRIELA MALENA QUISPE QUISPE
PROFESOR: ALEX FLOREZ FARFAN
AULA -8



UNSA
UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA

EJERCICIO 01:

Unique Paths:

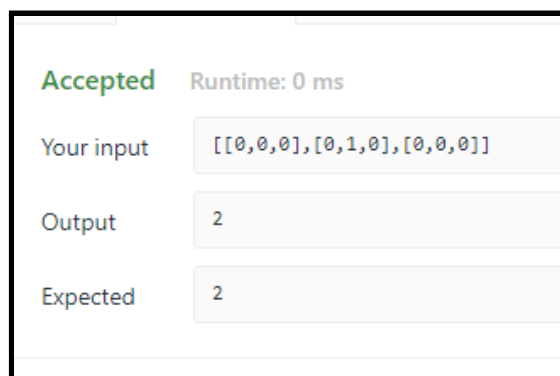
para este problema inicialmente creamos una matriz, luego cada celda representará el número de caminos que conducen a la celda correspondiente. Ya que el robot sólo puede moverse hacia la derecha o hacia abajo, podemos realizar una solución ascendente, trabajando desde la celda inicial e iterando hacia abajo y hacia la derecha.

Cada celda (obstáculo $[i][j]$) puede llegar a solo dos celdas visitadas previamente (obstáculo $[i-1][j]$ y $[i][j-1]$),

por lo que el número de formas para alcanzar la celda actual ($d[i][j]$) debe ser la suma de las formas de llegar a esas otras dos celdas (prueba $[i-1][j]$ + prueba $[i][j-1]$).

Resultados de las Pruebas:

Respecto a las pruebas, inicialmente me aceptó como se aprecia en la imagen para un tipo de entrada, y para otro tipo de entrada en la salida no me aceptaba.



Posteriormente modifique mi código y modifique en la página principal el mismo, el cual ya me aceptabas para las demás entradas.

Time Submitted	Status	Runtime	Memory		Testcase	Run Code Result	Debugger
12/03/2021 23:52	Accepted	0 ms	38.2 MB	j		Accepted	Runtime: 0 ms
12/03/2021 22:41	Accepted	0 ms	36.7 MB	j	Your input	[[0,0,0],[0,1,0],[0,0,0]]	
12/02/2021 18:32	Wrong Answer	N/A	N/A	j	Output	2	
					Expected	2	

Your previous code was restored from your local storage. [Reset to default](#)

Testcase	Run Code Result	Debugger
----------	-----------------	----------

Accepted

Runtime: 0 ms

Your input	[[0,0,0],[0,1,0],[0,0,0]]
Output	2
Expected	2

EJERCICIO 02:

$O(2^N)$ (costo del libro)
 max no posible
↓

$Book_1 \quad B_2 \quad B_3 \quad \dots \quad B_N, X$

Book-Shop

B_1 B_2 ... B_N
 $\swarrow \searrow$ $\swarrow \searrow$... $\swarrow \searrow$
 C_1 P_1 C_2 P_2 ... C_N P_N

4 10
 4 8 5 3
 5 12 8 1

$L = 10$
 costs: 4 | 8 | 5 | 3
 pages: 5 | 12 | 8 | 1
 $4 + 5 = 9$

1. $dp(i, x) \rightarrow \max^m$ numero: pages
 $i \rightarrow$ book
 x : money

2. $dp(N, x)$
 3. $dp(i, x) = \max \begin{cases} 1. dp(i-1, x) \\ 2. p_i + dp(i-1, x - c_i) \end{cases}$

Resultados de las Pruebas:

Tienda de libros	
TAREA	ENVIAR
RESULTADOS	ESTADÍSTICAS
HACKEAR	
Detalles de envío	
Tarea:	Tienda de libros
Remitente:	Gabriela Malena
Tiempo de presentación:	2021-12-03 03:05:28
Idioma:	C++ 11
Estado:	LISTO
Resultado:	ACEPTADO

Prueba 1

Veredicto: **ACEPTADO**

aporte
10 10 1 2 10 6 5 1 7 4 10 4 6 3 8 1 7 3 8 6 5 6

salida correcta
21

salida del usuario
21

Prueba 3

Veredicto: **ACEPTADO**

aporte
1000 1000 2 2 1 2 1 2 2 2 2 1 1 1 1 2 1 ...

salida correcta
747

salida del usuario
747

EJERCICIO 03:

El problema de la subsecuencia creciente más larga (LIS) es encontrar la longitud de la subsecuencia más larga de una secuencia dada de modo que todos los elementos de la subsecuencia se clasifiquen en orden creciente.

Si pensamos en cualquier subsecuencia, entonces puede comenzar desde cualquier lugar, por lo que para resolver esto, podemos aplicar un enfoque simple para corregir cada índice para que podamos calcular fácilmente el siguiente en cada índice final. Y a partir de esto podemos encontrar la relación de recurrencia.


Obteniendo:

$$\text{Lista}(\text{arr}, i) = 1 + \text{Lista}(\text{arr}, j) \rightarrow j < i \text{ and } \text{arr}[j] < \text{arr}[i].$$

Resultados de las Pruebas:

Testcase

Run Code Result

Debugger 

Accepted Runtime: 0 ms

Your input

[10,9,2,5,3,7,101,18]

Output

4

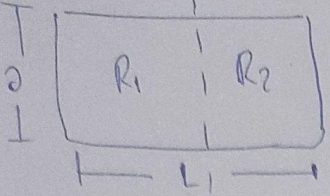
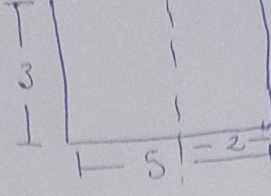
Expected

4

Time Submitted	Status	Runtime	Memory	Language
12/02/2021 20:29	Accepted	56 ms	38.5 MB	java

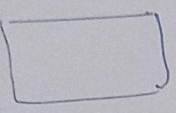
EJERCICIO 04:

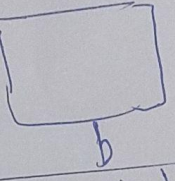
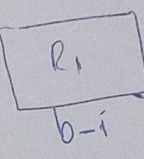
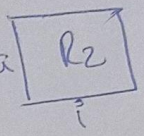
Rectangle Cutting

Embraces

- $dp(a, b) \rightarrow \min^m$ (cortes requeridos).
- $\rightarrow \sum \text{squares } (a \times b)$.
- $dp(a, b)$.
- Recurrencia

 $\rightarrow dp(a, a) = 0$

$dp(a, b) \rightarrow a$  $\rightarrow a$  

$$dp(a, b) = 1 + dp(a, b-i) + dp(a, i)$$

$$= 1 + dp(a-i, b) + dp(i, b)$$

Para el caso: $(i=1 \mid b=1) \mid a(A^3)$

Mínimo $\rightarrow (i=1 \mid a-1)$

Resultados de las Pruebas:

Corte de rectángulo



TAREA | ENVIAR | RESULTADOS | ESTADÍSTICAS | HACKEAR



Detalles de envío



Tarea:	Corte de rectángulo
Remitente:	Gabriela Malena
Tiempo de presentación:	2021-12-03 06:28:39
Idioma:	C ++ 11
Estado:	LISTO
Resultado:	ACEPTADO

Prueba 1

Veredicto: ACEPTADO



aporte	
2 8	 



salida correcta	
3	 



salida del usuario	
3	 

Prueba 2

Veredicto: ACEPTADO

aporte	
4 4	 

salida correcta	
0	 

salida del usuario	
0	 

EJERCICIO 05:

El problema requiere que el cuadrado contenga todos los 1 para que sea más grande. Cualquier cuadrado de esta matriz se define con tres parámetros:

x (fila),

y (columna) y también el

w (ancho).

Entonces para encontrar la matriz más grande, al menos buscaremos cada (x, y).

Por ejemplo, si todos los cuadrados son todos 1, el estado [3] [3] se convierte en $2 + 1 = 3$.

Entonces, el cuadrado más grande en el que termina (3, 3) no puede ser $2 + 1 = 3$. Además, no puede ser $1 + 1 = 2$ porque la condición 1, 2, 3 con 1 no son verdaderas. En resumen, derivamos la función de transición:

$$\text{problema}[x][y] = 1 + \min \{ \text{problema}[x-1][y], \text{problema}[x][y-1], \text{problema}[x-1][y-1] \}$$

Resultados de las Pruebas:

Testcase Run Code Result Debugger

Accepted Runtime: 0 ms

Your input

```
[["1","0","1","0","0"],["1","0","1","1","1"],["1","1","1","1","1"],["1","0","0","1","0"]]
```

Output

```
4
```




Expected

```
4
```

Next challenges:

Maximal Rectangle

Largest Plus Sign

Show off your acceptance:   

Time Submitted	Status	Runtime	Memory	La
12/03/2021 22:55	Accepted	4 ms	42.5 MB	ja