

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMOS AVANZADOS

**Primer Examen
(Primer Semestre 2024)**

Duración: 2h 50 min.

- **No puede utilizar apuntes, solo hojas sueltas en blanco.**
- En cada función el alumno deberá incluir, a modo de comentario, la forma de solución que utiliza para resolver el problema. De no incluirse dicho comentario, el alumno perderá el derecho a reclamo en esa pregunta.
- No puede emplear plantillas o funciones no vistas en los cursos de programación de la especialidad.
- Los programas deben ser desarrollados en el lenguaje C++. Si la implementación es diferente a la estrategia indicada o no la incluye, la pregunta no será corregida.
- Un programa que no muestre resultados coherentes y/o útiles será corregido sobre el 50% del puntaje asignado a dicha pregunta.
- Debe utilizar comentarios para explicar la lógica seguida en el programa elaborado. El orden será parte de la evaluación.
- Se utilizarán herramientas para la detección de plagios, por tal motivo si se encuentran soluciones similares, se anulará la evaluación a todos los implicados y se procederá con las medidas disciplinarias dispuestas por la FCI.
- **Solo está permitido acceder a la plataforma de PAIDEIA, cualquier tipo de navegación, búsqueda o uso de herramientas de comunicación se considera plagio por tal motivo se anulará la evaluación y se procederá con las medidas disciplinarias dispuestas por la FCI.**
- Para esta evaluación solo se permite el uso de las librerías **iostream, iomanip, climits cmath, fstream, vector, algorithm, string o cstring**
- Su trabajo deberá ser subido a PAIDEIA.
- **Es obligatorio usar como compilador NetBeans.**
- Los archivos deben llevar como nombre su código de la siguiente forma `codigo_EX1_P#` (donde # representa el número de la pregunta a resolver)

Pregunta 1 (10 puntos)

Se tiene un robot en un laberinto de $N \times N$ celdas, como el de la Figura. Los números de las celdas representan el costo de pasar por cada celda. Por ejemplo, 0=impasable, 1=tierra firme, 2=tierra arenosa, 3=lodo. El robot está ubicado en la celda (0,0) y se quiere que este llegue a la celda de salida (N-1, N-1). Desarrolle un algoritmo en C++ utilizando la técnica de *backtracking* para encontrar el camino de menor costo desde la celda (0,0) hasta la celda de salida (N-1, N-1). El robot puede hacer hasta 4 movimientos en cada celda: arriba, abajo, izquierda o derecha. Sin embargo, no son permitidos los movimientos que hagan salir al robot del laberinto o caer a una celda impasable.

1	2	0	1	0
3	2	3	1	1
0	1	2	0	0
3	1	1	2	3
0	1	3	1	1

Para la instancia de la figura de arriba (N=5), el camino de menor costo sería de costo 12, como muestra la figura de abajo. Asegúrese de probar su código en esta instancia.

1	1	0	0	0
0	1	0	0	0
0	1	0	0	0
0	1	1	1	0
0	0	0	1	1

Pregunta 2 (10 puntos)

La república de Bareim desea implementar un nuevo sistema monetario para manejar su economía interna, su moneda es el Dinar, por tal motivo debe decidir entre las diferentes denominaciones que podrían tener sus billetes o monedas, de tal forma que les sea útil a la población para llegar a una cantidad determinada de dinero. Para saber que tan útil son las denominaciones elegidas, decide traer desde Perú a un especialista en algoritmia para que desarrolle un programa que le ayude a determinar de cuantas formas puede llegar a un monto dado con las denominaciones planteadas. Por ejemplo:

Si se cuentan con las siguientes denominaciones: 2, 5, 10, 20 (expresado en dinares)
Se desea llegar al monto de 13 dinares

Respuesta: 1 forma (2, 2, 2, 2, 5)

Si se cuentan con las siguientes denominaciones: 2, 5, 10, 20 (expresado en dinares)
Se desea llegar al monto de 3 dinares
Respuesta: 0 formas

Si se cuentan con las siguientes denominaciones: 1, 2, 10, 20 (expresado en dinares)
Se desea llegar al monto de 7 dinares
Respuesta: 4 formas {(1, 1, 1, 1, 1, 1), (1, 1, 1, 1, 1, 2), (1, 1, 1, 2, 2), (1, 2, 2, 2)}

Desarrolle un algoritmo basado en programación dinámica que resuelva lo solicitado. Para obtener el puntaje de esta pregunta debe mostrar **la matriz o arreglo bidimensional** con las soluciones parciales, **en caso contrario su respuesta no será evaluada.**

Pregunta 3 (10 puntos)

a) La empresa Aceros Puno, se dedica a la comercialización de barras de acero de acuerdo con los cortes solicitados. Por política **no atiende pedidos parciales** y las dimensionan que les solicitan deben ser completas, en otras palabras, no se pueden formar de varios pedazos de barras. La empresa desea **maximizar la ganancia** al atender los pedidos, y debido a que muchas veces se solicitan barras con dimensiones mas pequeñas se pueden cortar las barras grandes en almacén para atender un pedido, **pero minimizando el desperdicio**, por ejemplo, si viene un pedido de 5 barras de 1 m. no se deben cortar 5 barras de 10 m. para obtener lo solicitado, sino tomar una barra de 10 m. y cortarla en trozos de 1m. lo que genera un desperdicio de 5 m. solamente. A continuación, se muestra un ejemplo de la atención de un conjunto de pedidos:

Si se tienen que atender **4** pedidos y en el almacén se cuentan con **20 barras de 10 m.**

Pedido	1	2	3	4	5
Dimensión	10	20	1	3	1
Cantidad	5	10	18	1	1
Ganancia (\$ K)	10	20	15	5	2

La respuesta será:

Ganancia: \$ 32 k

Desperdicio de barras utilizadas:

Barras	1	2	3	4	5	6	7	8	9	10	..
Desperdicio	0	1	0	0	0	0	0	7	10	10	..

Desarrolle un algoritmo heurístico que solucione el problema propuesto, mostrando la ganancia de atender los pedidos y el desperdicio originado por cada barra.

- b) Se tiene K máquinas de tejer (telares), cada una de las cuales puede operar a una velocidad diferente. Específicamente, los K telares pueden tejer 1m de tela en $T=\{T_1, T_2, \dots, T_k\}$ unidades de tiempo, respectivamente. Adicionalmente, se tiene N telas de diferentes tamaños (en metros) que deben ser tejidas, estos tamaños se denotan por $telas=\{t_1, t_2, \dots, t_N\}$.

Diseñe e implemente un algoritmo goloso en C++ para asignar la secuencia de telas que debe tejerse en cada telar de manera que se **minimice el tiempo total** necesario para tejer todas las telas, es decir, se minimice el tiempo en que el ultimo telar terminar su trabajo. Es importante notar que una tela no puede ser tejida en más de un telar ni dividirse entre varios telares.

Probar su implementación en la siguiente instancia: $K=3$, $T=\{5, 4, 6\}$, $N=6$, $telas=\{10, 7, 9, 12, 6, 8\}$

Tip: Como criterio goloso, puede asignar cada tela al telar que menos aumentaría el tiempo total. En ese caso, la salida del algoritmo para la instancia dada sería:

```
Asignaciones de telas a los telares:  
Telar 0: 7, 6 - Tiempo total de trabajo: 65 unidades de tiempo  
Telar 1: 10, 12 - Tiempo total de trabajo: 88 unidades de tiempo  
Telar 2: 9, 8 - Tiempo total de trabajo: 102 unidades de tiempo  
Tiempo total para terminar todas las telas: 102 unidades de tiempo
```

Al finalizar el laboratorio, comprima la carpeta de su proyecto empleando el programa Zip que viene por defecto en el Windows, **no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares**. Luego súbalo a la tarea programa en Paideia para este examen.

Profesores del curso:

Edwin Villanueva
Rony Cueva

San Miguel, 17 de mayo del 2024