

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

ALGORITMOS AVANZADOS

**1ra. práctica (tipo B)
(Primer Semestre 2025)**

Duración: 1h 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, 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_LAB1_P#` (donde # representa el número de la pregunta a resolver)

Pregunta 1 (10 puntos)

Considere usted que tiene un tablero de ajedrez de 8x8 casillas. Coloca la pieza del Rey en una de esas 64 casillas y le solicitan generar una ruta por la que el rey visite las restantes 63 sin pasar dos veces por la misma casilla. Considere lo siguiente:

- a) El problema tiene respuesta, o sea va a encontrar al menos una ruta que pase por todas las casillas del tablero.
- b) El rey se mueve en todas las direcciones posibles (8 direcciones), una sola casilla. No se confunda con el movimiento de la reina.

Se le pide desarrollar un programa en C++ que emplee backtracking para resolver el problema mencionado arriba pero que arroje la ruta que pueda recorrer el rey.

A continuación, un ejemplo colocando al rey en la casilla 4,4 del tablero, con una verificación horaria que parte de una diagonal.

Ruta encontrada:

4	5	6	7	8	9	10	11
24	3	22	20	18	16	14	12
25	23	2	21	19	17	15	13
52	26	27	1	29	30	31	32
53	51	49	28	39	37	35	33
61	54	50	48	40	38	36	34
62	60	55	56	47	41	42	43
64	63	59	58	57	46	45	44

Pregunta 2 (10 puntos)

Se tiene un almacén de dimensiones $n \times m$. Se desea llenar el espacio con palets de gran tamaño, considerando que el robot apilador pueda moverse sin problemas a través del almacén. Por tal motivo se debe tomar las siguientes consideraciones:

- Solo se puede colocar un palet en una ubicación de almacén
- Se puede colocar un palet en cualquier ubicación, excepto en la (0,0) ya que se reserva para la entrada al almacén
- Se puede colocar un palet al lado de las paredes del almacén o esquinas
- Alrededor de un palet no puede colocar otro palet, ya sea en horizontal, vertical o diagonal

Utilizando backtracking encuentre la cantidad máxima de palets que puede colocar en el almacén y la distribución dentro de él.

A continuación, un ejemplo en un almacén de 3×6 :

El maximo es:6

0	5	0	4	0	3
0	0	0	0	0	0
6	0	1	0	2	0

A continuación, un ejemplo en un almacén de 5×6 :

El maximo es:9

0	9	0	2	0	3
0	0	0	0	0	0
8	0	1	0	0	4
0	0	0	0	0	0
0	7	0	6	0	5

Profesores del curso:

Rony Cueva
Manuel Tupia
Igor Siveroni

San Miguel, 12 de abril del 2025