

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**ALGORITMIA**  
**Examen Final**

**(Primer Semestre 2020)**

Duración: 3 horas

**Nota:**

- Está permitido el uso de apuntes de clase y librerías desarrolladas en clase.
- Debe utilizar comentarios para explicar la lógica seguida en los programas elaborados, si en caso no se indica cómo ejecutar el programa desarrollado, no se revisará el código desarrollado.
- No se califica código comentado.
- Si la implementación es significativamente 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.
- Las variables y funciones deben tener nombres apropiados, en caso contrario no se revisará el código desarrollado.
- El orden será parte de la evaluación.
- Debe usar tipos de datos adecuados, según la pregunta lo requiera.
- Su trabajo deberá ser subido a PAIDEIA en el espacio indicado por los jefes de práctica.

Puntaje total: 20 puntos

---

**Elija 2 preguntas de entre las 3 preguntas propuestas:**

**P.1 (10 pts)**

Se tiene un almacén de dimensiones  $n \times m$ . Se desea llenar el almacén de palets, considerando que un 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

A. Desarrollo las estructuras necesarias y cargue los valores que crea conveniente en el almacén  $n \times m$ . (1 pto).

- B. Utilizando backtracking o programación dinámica encuentre la cantidad máxima de palets que puede colocar en el almacén (5 pts)
- C. Muestre la disposición de los palets dentro del almacén para una de las soluciones máximas (4 pts)

## **P.2 (10 pts)**

Una empresa de envío de paquetes realiza su servicio de manera específica: pueden enviar una caja de manera independiente o pueden juntar dos cajas para enviarlas como un solo paquete. La gerencia de la empresa está interesada en contar con un programa que les permita saber de cuántas formas es posible enviar una carga de  $n$  cajas, teniendo en cuenta que cada caja se puede enviar por separado o se pueden juntar las cajas en pares. Dado el número  $n$  de cajas, el programa debe imprimir en pantalla la cantidad de formas para enviarlas de acuerdo a las reglas de arriba.

Por ejemplo, si hay 3 cajas, existen 4 formas para enviarlas:

- {1}, {2}, {3}: Todas por separado
- {1}, {2, 3}: Primera caja sola y cajas 2, 3 en par
- {1, 2}, {3}: Cajas 1, 2 en par y la caja 3 por separado
- {1, 3}, {2}: Cajas 1, 3 en par y la caja 2 por separado

Consideraciones para la evaluación:

- A. Ingreso y salida de datos (1 pts)
- B. Implementación de algoritmo usando backtracking o programación dinámica (9 pts)

## **P3 (10pts)**

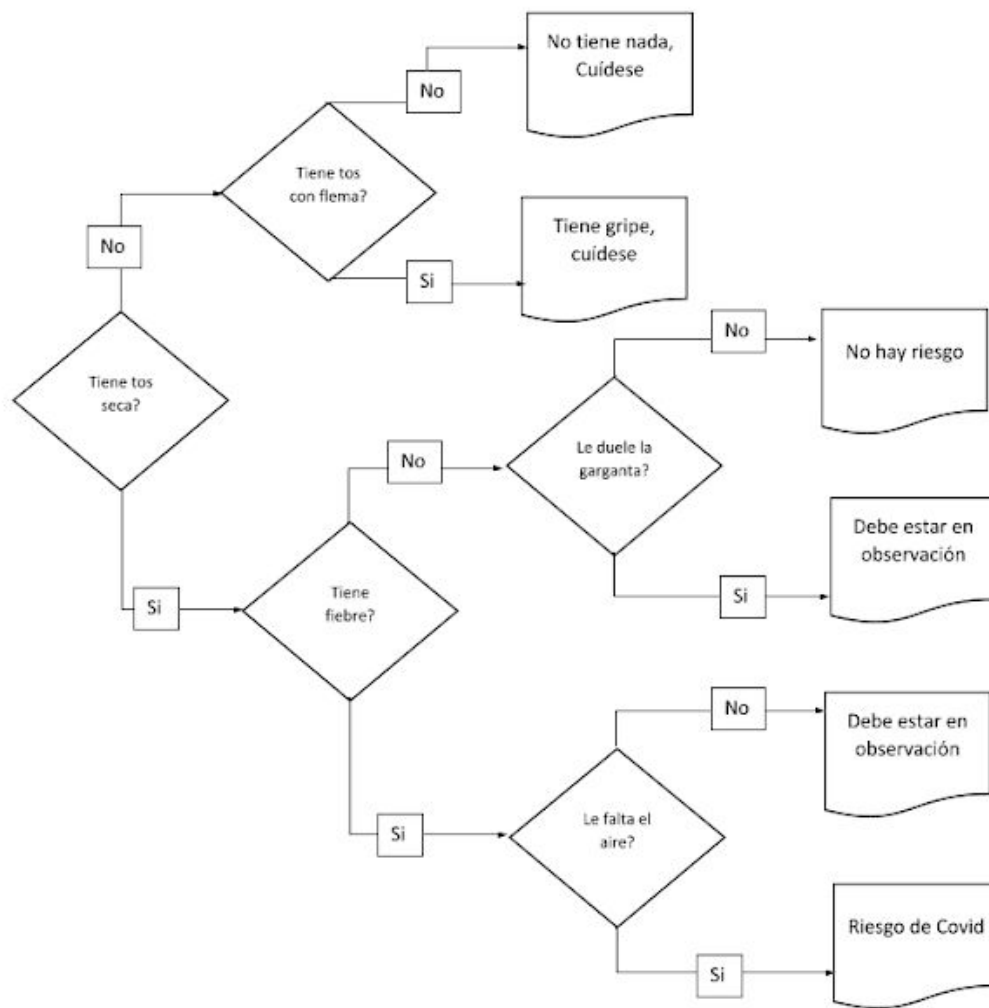
### **P3.1 (5pts)**

En los sistemas administradores de bases de datos, algunas operaciones se realizan sobre árboles binarios en los que las búsquedas se optimizan para funcionar en tiempo óptimo. Una forma de saber si un árbol binario de búsqueda es óptimo es midiendo la densidad de un ABB. La densidad de un ABB se calcula como la proporción entre el tamaño(número de nodos) y la altura(máximo número de nodos entre el nodo raíz y algún nodo hoja del árbol). La proporción se calcula como tamaño/altura.

A nivel de producción, los sistemas administradores de bases de datos desean realizar el cálculo de la densidad de la manera más rápida posible. El objetivo de esta pregunta es implementar una función que calcule la densidad de un ABB, con la condición de hacer un solo recorrido por el árbol.

### **P3.2 (5pts)**

Utilizando un árbol binario cargue el diagrama a continuación (debe cargar los valores fijos correspondientes a las preguntas y respuestas que se muestran). Como se observa en el gráfico los nodos que tienen hijos son preguntas y los nodos finales que no presentan hijos son las respuestas a las que se debe llegar. Desarrolle una función que muestre cada pregunta y según su respuesta Si(S) o No (N) avance en el árbol binario, hasta finalmente mostrar una respuesta con el diagnóstico de los síntomas.



**Profesores del Curso:**

- Rony Cueva
- Johan Baldeón
- Iván Sipirán

San Miguel, 01 de agosto del 2020