

## **PEP 2 Paradigmas de Programación (2021-2)**

**NOMBRE:** \_\_\_\_\_ **RUT:** \_\_\_\_\_ **Profesor:** \_\_\_\_\_

### **Instrucciones**

- Tiempo estimado de respuesta: 90 minutos
- Tiempo disponible para entregar respuestas desde que tuvo acceso al enunciado: 3 horas hasta las 23:59 del 23 de Diciembre de 2021.
- Lugar de entrega: Campus Virtual.
- Escriba sus respuestas en hojas separadas
- Puede escribir sus respuestas directamente en un documento electrónico o bien puede hacerlo en papel. Escoja el que más le acomode.
- Recomendación: Para asegurar acceso continuo a este enunciado en caso de cortes de suministro eléctrico o en el acceso a Internet, procure descargar este enunciado y si puede, imprímalo.
- Identifique cada hoja de respuestas con su nombre y rut, lo que sirve de complemento a la declaración que ha aceptado en Campus Virtual.
- Identifique cada hoja de respuesta con el profesor de su sección.
- Al terminar la evaluación,
  - Si desarrolló sus respuestas con lápiz y papel, saqué una fotografía o escanee cada una de sus hojas de respuestas. Asegúrese que las imágenes de sus capturas son legibles (abra el archivo y verifique). Luego, agrupe todas las imágenes en un PDF (en Windows basta con seleccionar las imágenes desde el explorador de archivos -> clic derecho -> Imprimir -> Escoger Impresora PDF de Microsoft u otra que disponga) o bien en un archivo comprimido .zip o .rar.
  - Si desarrolló sus respuestas directamente en formato digital (ej: documento Word, Google Docs, Latex, etc.) guardar el documento como PDF. Se recomienda este formato para evitar problemas en la visualización de estilos.
- Finalmente, subir el archivo al espacio creado para tales efectos en Campus Virtual. Una vez subido el archivo, descárguelo y asegúrese que el archivo se puede abrir (para evitar situaciones de archivos dañados).
- Excepción: En la eventualidad que no pueda subir su trabajo a Campus Virtual en el plazo disponible, enviarlo inmediatamente vía correo a su profesor.

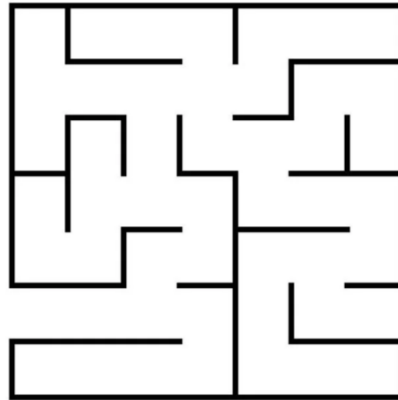
**Sobre la evaluación. Todo ítem se evalúa en una escala de 3 puntos: 0 (sin respuesta o respuesta no aborda el problema), 0.5, 1 (respuesta completa sin errores), según grado de cumplimiento.**

**NOTA 1:** Procure hacer un adecuado uso de la programación declarativa y particularmente de la programación lógica. La implementación innecesaria y/o redundante de predicados que desaprovechan el paradigma lógico tendrán una penalización de hasta **2 puntos** por cada ítem afectado.

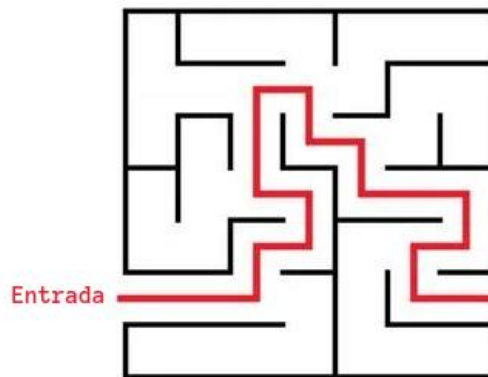
**Nota 2:** Especificar e implementar solo lo necesario de los TDAs que sean requeridos para abordar el problema.

**Nota 3:** Documentar y estructurar programa considerando: dominios, predicados, metas y cláusulas.

1. **(30 pts)** Un laberinto es un lugar formado por calles y encrucijadas predispuestas de tal forma de confundir a quien se adentre en él. Los más antiguos son de forma cuadrada y/o rectangular, actualmente son un buen juego de ingenio para niños e incluso los más adultos cuando son lo suficientemente grandes en sus dimensiones.



Por ejemplo para el laberinto anteriormente señalado, si se comienza por la entrada del lado izquierdo ubicada en la fila 6, columna 1 (suponiendo que lo dividimos en cuadrantes) el camino de salida sería el siguiente:



- Indique una representación mediante hechos para cualquier laberinto de forma rectangular **(3 pts)**.
- Expresa el laberinto indicado anteriormente en las imágenes utilizando su representación **(5 pts)**.
- Desarrolle un predicado en prolog que permita determinar si un laberinto tiene un camino que permita llegar a la salida a partir de una posición de entrada **(7 pts)**.
- Desarrolle un predicado en prolog que permita relacionar una entrada del laberinto, una salida del laberinto y la ruta que debe seguirse para esa ruta **(15 pts)**.

2. (30 pts) En el contexto de la elección de los Consejeros Académicos por Facultad de la Universidad Autónoma de Leonera (UAL), la Federación de Estudiantes le solicita a usted construir un programa computacional en pseudo-Prolog que ayude a consultar los resultados recientemente obtenidos en la elección. Para ello se le entregan los siguientes datos de muestra para realizar su trabajo:

ID Candidato	Nombre Candidato	Lista	Facultad
1	Elsa Paio Grandón	PCO	FI
2	Nicolás Schaefer Negrete	ELP	FI
3	Diego Achurra Pinto	LRA	FI
4	Zacarías Labarca del Río	PCO	FH
5	Alarico Rojas Vallenar	LRA	FH
6	Antonia Hoffmann Pérez	ELP	FE
7	Casimiro Vásquez Ponce	LRA	FE
8	Rosamel Vallejos Zúñiga	PCO	FE

ID Candidato	Votos
1	1283
2	368
3	669
4	934
5	612
6	715
7	35
8	710

ID Lista	Nombre Lista
PCO	Partido Campesino Obrero
ELP	Estudiantes Liberales por el Progreso
LRA	Lista Radical por la Autogestión

ID Facultad	Nombre Facultad
FI	Facultad de Ingeniería
FH	Facultad de Humanidades
FE	Facultad de Economía

Considerando estos datos, se le solicita desde la Federación de Estudiantes de la UAL (FEUAL) que su programa pueda realizar las siguientes operaciones:

1. Expresar los datos de muestra como una base de hechos en pseudo-Prolog, considerando una representación apropiada para estos datos **(5 pts)**.
2. Crear un predicado que dado el nombre de una Lista (ej: "Estudiantes Liberales por el Progreso") entregue como resultado de la consulta el nombre de todos sus candidatos, el nombre de la Facultad a la que pertenecen y sus votos obtenidos **(7 pts)**.
3. Crear un predicado que dado el nombre de una Facultad (ej: "Facultad de Humanidades") entregue como resultado de la consulta el número total de votos válidamente emitidos en dicha Facultad **(8 pts)**.
4. Según el reglamento de la FEUAL, el candidato mayoritario puede ser declarado como vencedor si posee el 50% o más del total de votos válidamente emitidos en su Facultad. Construya el/los predicado(s) que permitan obtener los candidatos vencedores de la elección (señalando nombre candidato, nombre facultad, nombre lista y número de votos) de acuerdo a esta condición **(10 pts)**.