

Laboratorio 1: Prolog

Condiciones de entrega

- Trabajo grupal.
- Fecha de entrega: 12 de abril.
- Forma de entrega: archivo zip conteniendo el código fuente del programa y un documento (pdf) donde se detalle:
 - Autores del trabajo
 - Explicación breve de cómo funciona la solución
 - Si existen bugs o problemas conocidos al momento de la entrega
 - Cómo se ejecuta (qué consulta hay que realizar),
 - Cualquier cosa que consideren importante detallar al momento de la entrega.

El archivo deberá ser subido al aula virtual en la zona de entrega titulada “Entrega Laboratorio 1”.

Condiciones de aprobación

Es necesario para la aprobación del trabajo que

- el programa compile correctamente
- el programa brinde por lo menos la posibilidad de realizar las consultas solicitadas en la "Fase 1" del enunciado
- se adjunte la documentación requerida

Criterio de evaluación

Se tendrá en cuenta los siguientes criterios al evaluar el trabajo.

- Las fases del trabajo que fueron completadas.
- Si el programa genera *warnings* o tiene variables en sin utilizar.
- Demás requisitos definidos en el trabajo
- Adecuación del programa al paradigma. Uso de la recursión, el pattern matching, manejo del backtracking y demás herramientas vistas en clase para la resolución del problema.
- Claridad de la documentación aportada.

Enunciado

La dirección de una empresa de repartos solicita al área de soporte informático un programa que le permita distribuir entre sus empleados los pedidos de trabajos a realizar, de acuerdo con sus capacidades. Para ello aporta la siguiente información, a fin de volcar la misma en la base de conocimiento:

- De cada empleado se registra su *nombre* y las *capacidades* con las que cuenta, las que permiten determinar si es apto o no para desarrollar un trabajo de un tipo determinado.
- De cada tipo de trabajo se registra su *identificador* y la lista de *capacidades* con las que debe contar un empleado para poder realizarlo.

La dirección indica que, para comenzar a trabajar, dispone de los siguientes datos:

- Tipos de trabajos
 - repartir mensajes/paquetes en el radio urbano
 - repartir mensajes/paquetes a larga distancia
 - clasificar paquetes
 - realizar logística
 - atención al público
- Lista de empleados
 - Carlos: disponibilidad horaria, fuerza
 - Marisa: fuerza, buena atención
 - Juan: movilidad, disponibilidad horaria
 - Jimena: logística, disponibilidad horaria
 - Héctor: buena atención, logística, fuerza
 - Betty: movilidad, buena atención, disponibilidad horaria
 - Lucía: buena atención
 - Axel: fuerza, movilidad
 - Eva: movilidad, logística
 - Miguel: buena atención, disponibilidad horaria
 - Clara: fuerza
 - Luis: logística

Es necesario tener en cuenta que:

- repartir requiere *fuerza*, y si es en larga distancia requiere además *movilidad*.
- realizar logística requiere saber de *logística*
- clasificar paquetes requiere *disponibilidad horaria*
- atención al público requiere *buena atención*

Fase 1

- Diseñar una base de conocimiento adecuada, aplicando extensivamente funtores y listas.
- Desarrollar las siguientes reglas:
 - `verifica(Empleado, TipoTrabajo)` tal que dado un empleado y un tipo de trabajo verifica que el empleado está capacitado para realizarlo.
 - `puedenRealizar(idTrabajo, Empleados)` tal que dado un identificador de trabajo a realizar permite determinar la lista de empleados capacitados para realizarlo.
 - `asignarTrabajos(Trabajos, Asignaciones)` tal que dada una lista de trabajos a realizar, permite obtener todas las combinaciones posibles de lista de asignaciones, donde cada elemento relaciona cada trabajo con un empleado que puede realizarlo, teniendo en cuenta sus capacidades y que se le puede asignar sólo un trabajo por vez.

Fase 2

- `asignarTrabajosGrupales(TrabajosGrupales, GruposAsignados, TrabajosARechazar)` tal que dada una lista de trabajos grupales a realizar, donde cada uno determina además la cantidad de empleados requeridos para su realización, permite obtener todas las combinaciones posibles de la lista de grupos asignados. Cada elemento de la lista de grupos asignados relaciona cada trabajo con los empleados que pueden

realizarlo teniendo en cuenta sus capacidades, que se le puede asignar sólo un trabajo por vez a cada empleado, y que se debe contar con la cantidad de empleados necesaria. La lista de trabajos a rechazar determina los trabajos que no sería posible realizar.

- Ejemplificar todas las posibles consultas que el sistema desarrollado soporta, analizando la posibilidad de inversibilidad de cláusulas.
- Analizar la posibilidad de definir reglas con recursividad de cola.