

## **PARTE 2 (en equipos): Ejercicios de aplicación**

**Duración: 45 minutos – sin material**

### **EJERCICIO 1**

Estamos considerando estudiar una cierta carrera, y nos interesa conocer las diferentes secuencias de asignaturas que deberemos cursar.

Como en todas las carreras, las asignaturas suelen requerir la aprobación de otras asignaturas a las que llamamos “previas”.

Conocemos la información de cada materia: Nombre (ej: “Cálculo”) y duración (ej.: 1 semestre).

También se nos ha brindado una lista de “previaturas”, en que se indican, para cada asignatura, todas aquellas otras que es necesario aprobar previamente para poder cursarla.

Para simplificar, podríamos asumir que existe un estado “**Ingreso**”, (con **duración 0**, único que no tiene “previas”), y también un estado “**Graduación**” (también **duración 0**, y que no es “previa” de ninguna materia).

Deseamos saber también cuál será la “**secuencia crítica de cursos**”, es decir, aquélla, **entre todas las secuencias posibles (desde “Ingreso” hasta “Graduación”)** en la cual cualquier retraso (en cualquiera de los “cursos” que la componen) determinará que completar la carrera nos lleve más tiempo (ver diagrama ejemplo al reverso).

Se requiere entonces:

1. Indicar cómo podemos representar este problema. Ilustrar con un diagrama sencillo.
2. Desarrollar un algoritmo que devuelva la secuencia de cursos crítica (listado de los cursos componentes, cada cual con su duración) y el tiempo total mínimo que insumiría la misma.
3. ANALIZAR EL ORDEN DEL TIEMPO DE EJECUCIÓN. ES IMPORTANTE DISEÑAR UN ALGORITMO CON EL MEJOR ORDEN POSIBLE
4. ¿Qué precauciones debemos tomar con el conjunto de datos que representa el problema, para que nuestro algoritmo funcione correctamente?

29 de octubre de 2015

