

## Aplicaciones Computacionales en Negocios

### Trabajo Práctico 2 - Noviembre de 2025

---

Se viene el período de exámenes en la universidad, y tenemos que definir la programación de los segundos parciales. El problema consiste en determinar en qué día y horario se debe tomar cada parcial, respetando diversas restricciones. Tenemos un conjunto  $P$  de parciales, que se deben programar entre el 1/12 y el 12/12 (excepto el 6/12, 7/12 y 8/12, que corresponden a un fin de semana y un feriado, respectivamente). Cada parcial se puede programar a las 9:00, a las 12:00, a las 15:00 o a las 18:00, y suponemos que ningún parcial tiene una duración mayor a tres horas. Tenemos además un *grafo de incompatibilidad*  $G = (P, E)$  entre parciales, cuyos vértices representan los parciales y cuyas aristas unen pares de parciales que tienen estudiantes en común, y entonces no se pueden programar en el mismo día y horario. Finalmente, para cada parcial  $p \in P$ , tenemos la cantidad  $a_p \in \mathbb{Z}_+$  de aulas que ese parcial requiere.

En este contexto, el problema consiste en asignarle un día y un horario a cada parcial, de modo tal que se respeten las siguientes restricciones.

- Si  $pq \in E$ , entonces los parciales  $p$  y  $q$  no pueden estar asignados en el mismo día y horario (aunque sí pueden estar asignados en el mismo día pero en distintos horarios).
- La cantidad total de aulas utilizadas en cada momento no debe superar las 75 aulas disponibles. Es importante notar que si un parcial requiere de varias aulas, entonces se toma el parcial al mismo tiempo en todas las aulas asignadas. En este momento no es relevante asignar aulas específicas a los parciales, sino que alcanza con saber que las aulas disponibles serán suficientes.

No siempre será posible programar todos los parciales cumpliendo las restricciones, y en este caso buscamos maximizar la cantidad de parciales programados. El resto de los parciales se deberá programar manualmente o en fechas/horarios adicionales.

1. Proponer un modelo de programación lineal entera para maximizar la cantidad de parciales que se pueden asignar respetando las restricciones. Aplicar el modelo a los datos proporcionados junto con este enunciado, y analizar la solución obtenida.
2. ¿Cómo se modifica el modelo si ahora buscamos impedir que haya estudiantes rindiendo tres parciales un mismo día? (es decir, no queremos que haya tres vértices vecinos en  $G$  programados un mismo día) ¿Se puede resolver este problema en tiempos de cómputo razonables? ¿Se modifica la cantidad máxima de parciales que se pueden programar?
3. Supongamos que tenemos un peso asociado con cada arista  $pq \in E$ , que especifica la cantidad de estudiantes en común que tienen los parciales  $p$  y  $q$ . ¿Cómo se modifica el modelo si ahora queremos maximizar la *dispersión* de los parciales entre los estudiantes? Es decir, queremos evitar que un mismo estudiante tenga todos sus parciales concentrados en pocas fechas consecutivas. Notar que este problema no está especificado con

precisión, y entonces parte de nuestro trabajo es proponer una función objetivo adecuada para tratar esta situación. Además, no tenemos los datos de inscriptos en cada sección sino solamente la cantidad de estudiantes en común entre cada par de secciones. ¿Qué podemos hacer con esta información limitada?

**Condiciones de entrega:** El trabajo práctico se debe entregar a través del campus virtual. La entrega debe incluir un breve informe detallando el o los modelos planteados, junto con todos los archivos necesarios para ejecutar el modelo. El trabajo práctico se puede hacer en grupos de hasta cuatro personas.

**Fecha de entrega:** Lunes 24 de noviembre.

**Formato de los archivos de datos.** Junto con este enunciado, se entregan tres archivos con los datos necesarios para resolver el problema, de acuerdo con la siguiente descripción:

1. El archivo `cursos.dat` contiene los IDs de los cursos en la primera columna, y la cantidad de aulas que requiere cada curso en la segunda columna.
2. El archivo `estudiantes-en-comun.dat` contiene tres columnas: las primeras dos columnas especifican los IDs de dos cursos vecinos en  $G$  y la tercera columna especifica la cantidad de estudiantes en común entre esos dos cursos. Los pares de cursos que no están en este archivo no tienen estudiantes en común.