

EXAMEN PARCIAL 2

Enunciado

Un conocido colegio secundario de Buenos Aires que cuenta con 8 sedes y nos pide desarrollar un programa que los ayude a procesar la información de las solicitudes de inscripción de los alumnos a ingresar en el próximo ciclo lectivo.

De las ocho sedes existentes, hay dos tipos: propias y tercerizadas.

Todas las sedes tienen un nombre, una dirección y la cantidad siempre actualizada de vacantes disponibles por turno (mañana, mediodía, tarde o noche).

Hay una diferencia en la forma en que las sedes propias o tercerizadas mantienen y guardan las solicitudes: mientras que en las sedes propias las solicitudes son guardadas en orden descendente por promedio de notas del alumno, las sedes tercerizadas las solicitudes se guardan en el mismo orden en el cual se recibieron.

Otra diferencia es que las sedes tercerizadas tienen el nombre de la institución que la respalda.

Cada solicitud debe ser procesada en el mismo orden en que se recibió, teniendo de esta forma prioridad los primeros en anotarse.

De cada una de las solicitudes a procesar conocemos la siguiente información:

- Nro de solicitud: Entero
- Alumno
- Nombre de la sede elegida para cursar: String
- Turno elegido (mañana, mediodía, tarde o noche)
- Confirmada (inicialmente en falso)

De cada uno de los posibles alumnos sabemos:

- Número de documento
- Nombre
- Promedio de notas.
- Zona de residencia: Entero 1 a 10

Se pide:

- Diseñar el diagrama UML completo que resuelva el problema propuesto.
- Desarrollar en NS+ los siguientes puntos (incluyendo los métodos derivados que utilices):
 - Desarrollar un método llamado **procesarSolicitudes** que para cada solicitud a procesar consulte si para la sede solicitada y el turno solicitado hay vacantes. En el caso que haya se debe confirmar la solicitud y descontar la vacante utilizada. Si la sede es propia, debe guardar la solicitud en una lista de espera de la sede ordenada por promedio de nota del alumno. Y si es una sede tercerizada, debe guardarla en una lista de espera de la sede en el mismo orden que se procesó. Al finalizar este proceso, quedarán en la estructura original sólo las solicitudes que no fueron confirmadas, es decir, para las que no hubo vacante.
 - Desarrollar un método llamado **informarDisponibles** que muestre por pantalla y para cada turno cual es la sede que tiene más vacantes disponibles.
 - Desarrollar un método llamado **informarEstadistica** que informe por pantalla la cantidad de solicitudes a procesar que hay para cada sede y zona de residencia. Por ejemplo:

Sede Moron:

 Zona 1: 4 solicitudes

 Zona 2: 3 solicitudes

 ...

 Zona 10: 2 solicitudes

Sede San Justo:

Zona 1: 15 solicitudes

Zona 2: 8 solicitudes

...

Zona 10: 5 solicitudes

Y así sucesivamente para todas las sedes y zonas.