



Proyecto 2 Enero – Marzo 2024

Pide Cola

1 Introducción

La Universidad Simón Bolívar actualmente busca regresar a la presencialidad. Existen dos preocupaciones principales: ¿Cuáles estudiantes podrían asistir? ¿Cuáles cursos se podrán dictar?

Para evaluar la factibilidad de la propuesta, se le pide que desarrolle un programa que incorpore los siguientes supuestos:

- Todo estudiante de la USB tiene un 15% de probabilidad de poder montarse en un autobús de la universidad con el servicio actual
- Todo estudiante tiene un 10% de probabilidad de poder venir en transporte público
- La probabilidad de que un estudiante pueda venir en cola depende de cuántos otros estudiantes conozca con carro
- Entre más años tenga un estudiante en la universidad, mayor es la probabilidad de que tenga carro
- La probabilidad da la proporción de los estudiantes

Para ello, DACE le ha dado acceso un directorio que contiene todos los comprobantes de inscripción y listas de clase.

2 Requerimientos del programa

Diseñe e implemente un programa que pueda ser llamado de la siguiente manera:

```
$ ./probAsistencia <carnet|codigo> [-c|--cohortes <n>] [-d <dir>]  
[-p|--pcohortes <m>] [-i|--incremento <q>]
```

donde

- `carnet` es el carnet del estudiante a analizar, o alternativamente
- `codigo` es el código de curso a analizar;
- `n` es la cohorte que más recientemente ingresó en la universidad (por defecto la cohorte 21),
- `d` es el directorio de DACE que contiene todos los comprobantes de inscripción y listas de clase (por defecto el directorio se llama "DACE"),
- `m` es la probabilidad porcentual de que un estudiante de la cohorte `n` tenga un carro (por defecto, se debe establecer en 3% – note que `p` podría no ser entero) e
- `q` es el aumento en la proporción de carro-habientes entre la cohorte `n` y la cohorte `n – 1` (por defecto el incremento debe ser de 4 puntos porcentuales). Suponga que esta diferencia se mantiene entre cualquier par de cohortes.

2.1 Formato de archivos

Se proporcionan dos tipos de archivo, comprobantes de inscripción y listas de clase

Los comprobantes de inscripción llevan como nombre de archivo `<carnet>.txt` y tienen el formato

```
<carnet>
<códigoi> seccion: <seci> <nombrei>
...
<códigon> sección: <secn> <nombren>
```

Donde `códigoi`, `sec` y `nombrei` corresponden al código, sección y nombre de un curso en el cual se inscribió el estudiante `<carnet>`

Las listas de clase llevan como nombre `"<código> seccion <sec>.txt"` y tienen el formato

```
<Código> seccion: <sec> <nombre> prof. <profesor>-<sede>
<carnet1>
...
<carnetm>
```

Donde `carneti` es cada uno de los carnets de los estudiantes inscritos en el curso

2.2 Estructura de directorios

Todos los archivos se agrupan primero por sede. Los comprobantes se agrupan por cohorte y las listas de clase, por departamento. Los comprobantes se encuentran separados de las listas de clase como se muestra en la Figura 1.

La cohorte viene dada por los primeros dos dígitos del número de carnet. El departamento viene dado por las primeras dos letras del código del curso.

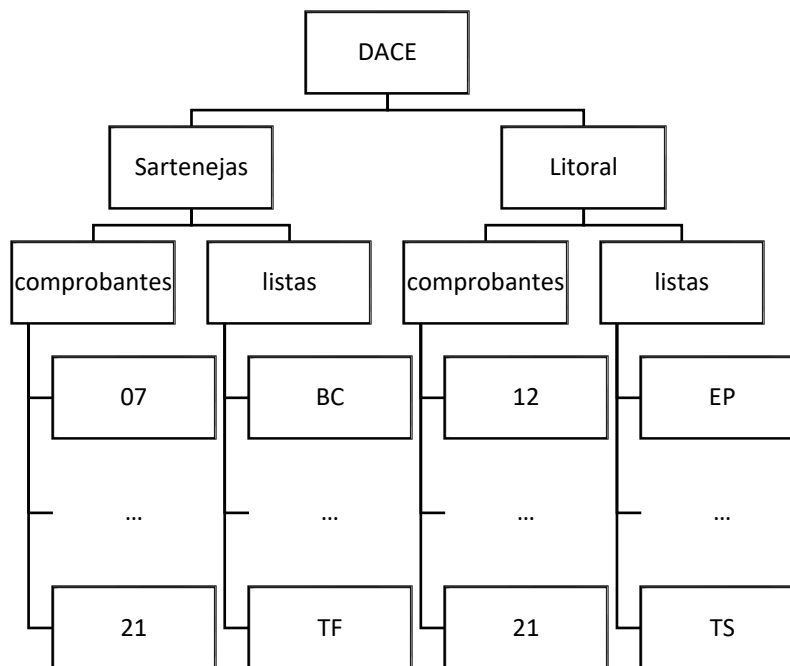


Figura 1. Árbol de directorios para el trimestre Enero-Marzo 2024

2.3 Funcionamiento del programa

2.3.1 Consulta por Carnet

El programa debe recibir el carnet y navegar, desde el directorio raíz dado (o el directorio “DACE” en el actual si no se da un directorio raíz) hasta el comprobante respectivo. Este tendrá la lista de cursos que el estudiante preinscribió. Para cada uno de ellos, el programa deberá navegar a la lista de clases respectiva, las cuales contienen los estudiantes que se inscribieron. Tome en cuenta que existen estudiantes de Sartenejas que toman cursos de El Litoral.

Debido a que los comprobantes se emiten antes de las listas de clase, el programa debe contemplar que el estudiante puede haber corregido inscripción para retirarse del curso: debe revisar que, en efecto, el estudiante aparece en la lista de cada curso en su comprobante. No es necesario considerar que el estudiante agregue cursos en corrección (si se hace, será considerado para puntos extra con su debida justificación en el informe).

Si el estudiante aparece en lista, todos los demás estudiantes pasan a ser posibles candidatos para darle la cola. Para encontrar la probabilidad de que un estudiante le dé la cola, se debe sumar la probabilidad dada por su cohorte. Así, si un curso tiene 10 estudiantes cohorte 21 (la cohorte más reciente), 5 estudiantes cohorte 20 y 2 estudiantes cohorte 19 (excluyendo al estudiante por el cual se está haciendo la consulta), la probabilidad de que el estudiante consiga cola en ese curso es de

$$3\% \times 10 + 7\% \times 5 + 11\% \times 2 = 87\%$$

Aunado a la probabilidad de venir por otros medios, el estudiante llega al 100% (nótese que no se puede exceder el 100%)

En caso de no alcanzar o exceder el 100%, se le debe sumar las probabilidades de los otros cursos que está cursando.

2.3.2 Consulta por código de curso

El programa deberá buscar la lista de clase de ese curso, y calcular la proporción de estudiantes que tienen carro. Si esa cantidad es mayor que un tercio del número de inscritos en el curso, luego de restarle el 10% y 15% que viene en transporte público y USB. Se debe imprimir, entonces la cantidad de carros esperados en el curso, la cantidad de inscritos, y si los carros son suficientes.

En el ejemplo anterior, 10 estudiantes cohorte 21 son 0,3 carros; 5 estudiantes cohorte 20 son 0,35 carros, y 2 estudiantes cohorte 19 son 0,22 carros, para un total de 0,9 carros. El total de inscritos es de 17 estudiantes y luego de restar el 10% y 15% de eso quedan 12 estudiantes, los cuales requerirían 4 carros. Ya que sólo hay 0,9 carros disponibles, los carros no son suficientes.

3 Informe

Debe entregar un informe que contenga

- Introducción y estructura del informe
- Decisiones de diseño e implementación
- Análisis para los miembros del grupo
- Análisis para las materias cursadas por los miembros del grupo
- Conclusiones y recomendaciones para la universidad y desarrolladores futuros

4 Requerimientos de la entrega

Debe entregar un archivo `tar.gz` que contenga

- Los archivos de código de su programa
- Makefile
- Su informe

El proyecto debe ser subido al Moodle de la materia en la sección marcada como “📁 Proyecto 2” en la sección de “semana 8”. Sólo deberá efectuar una entrega por grupo.

5 Evaluación

El proyecto tiene una ponderación de 15 puntos. Se asignarán

- 5 puntos por código
 - 2 puntos por sus llamadas a `dirent`, `stat` y/o `readdir`
 - 1 punto por su forma de recorrer los archivos
 - 1 punto por su forma de calcular la probabilidad de un estudiante de conseguir cola
 - 1 punto por su forma de calcular la cantidad de carros disponibles en un curso
- 5 puntos por ejecución
 - 2 puntos por su manejo de los flags
 - 2 puntos por imprimir correctamente la probabilidad de un estudiante de conseguir cola
 - 1 punto por imprimir correctamente la cantidad de carros disponibles en un curso
- 5 puntos por informe
 - 1 punto por su Introducción y estructura del informe
 - 1 punto por sus Decisiones de diseño e implementación
 - 1 punto por su Análisis para los miembros del grupo
 - 1 punto por su Análisis para las materias cursadas por los miembros del grupo
 - 1 punto por sus Conclusiones y recomendaciones

El programa debe correr sin errores.

5.1 Puntos Extra

Se asignarán puntos extra si

- Se busca en qué cursos se inscribió el estudiante luego de generarse el comprobante
- Se buscan los carros disponibles para los estudiantes del curso a través de los otros cursos que están cursando
- Se desarrolla un *shell script* que corra el programa para todos los estudiantes
- Se desarrolla un *shell script* que corra el programa para todos los cursos
- Cualquier otra funcionalidad adicional justificada en el informe