

## Ejercicios propuestos

Cada uno de los ejercicios propuestos se recomienda hacerlo en parejas o en grupos de a tres alumnos. De manera adicional, se recomienda que repasen todo el contenido de las diapositivas, ejercicios, entre otros, con el fin de poder resolverlos. Importante destacar es que los problemas representan ejercicios de complejidad creciente y tratan de abarcar todo el contenido de la sección Python. Además, es posible visualizar este tipo de problemas como ejercicios de bases de datos de documentos sin emplear un gestor de bases de datos, lo cual le entrega una mayor complejidad.

Se espera que los alumnos puedan resolver estos problemas en tres semanas y que usen los horarios de clases para resolver las preguntas y cuestionamientos ante los mismos.

### Decidiendo si me hago un auto regalo

Al regresar de su campaña sienten que se merecen un premio, razón por la cual, haciendo usos de sus ahorros, deciden que quieren comprarse algo, obviamente dicha compra depende del presupuesto, la disponibilidad y el tiempo de demora que tarde el delivery en hacer la entrega de su producto. Con el fin de apoyar en esta decisión, usted decide armar un script que facilite indicar el tipo de producto que desea comprar, según su presupuesto y las condiciones expuestas. Para ello, se tomará como referencia a Juanito Castro, quien tiene las siguientes condiciones:

- Si el producto tiene un valor mayor a \$20.000 lo comprará sin importar la espera del tiempo de entrega.
- Si el producto tiene un valor entre \$20.000 y \$15.000 lo comprará sí y sólo sí el tiempo de espera es menor a 5 días.
- Si el producto tiene un valor menor a \$15.000 lo comprará si el tiempo de entrega es menor a 3 días.
  - Identifique las entradas del script
  - Identifique las salidas y el proceso
  - Implemente el script con las condiciones propuestas
- Modifique el script anterior para que las condiciones las pueda procesar por la entrada estándar.
- Dado el script realizado en el punto 1, modifíquelo para que trabaje con un número determinado de estudiantes, manteniendo las condiciones del punto 1.
- Implemente una función que dada las condiciones decida si compra o no el producto
- Dado el script realizado en el punto 3, reciba los datos, presupuesto y nombre de alumnos en un archivo de texto y manteniendo las condiciones impuestas en el primer problema, determine la decisión de cada alumno.

***Hint 1: Use la función implementada en el punto 4 para trabajar con la decisión de compra.***

**Hint 2:** Para el archivo, se recomienda utilizar una estructura nombre-presupuesto, es decir, emplear un “-” para separar ambos valores

## Estimando el rendimiento de los alumnos

Uno de los problemas clásicos a los que uno se enfrenta es estimar el rendimiento de un alumno, normalmente, esto se logra normalmente en una planilla de cálculo. Sin embargo, el hecho de mantenerla actualizada y trabajar con ella es una tarea rutinaria que puede optimizarse según los requerimientos. Dado a sus nuevas competencias computacionales como desarrollador de scripts, decide por iniciativa propia implementar un script que facilite la estimación del rendimiento de los alumnos para un listado de ramos. Dado lo siguiente se plantea que este script pueda cumplir con los siguientes puntos.

1. Reciba un archivo con la lista de las notas por ramo con el fin de leerlas, este archivo debe tener el siguiente formato de nombre listado\_de\_materias.txt. Además, el contenido del archivo debe ser el siguiente

materia1.txt

materia2.txt

.  
.  
.

2. Cada archivo de los indicados en el punto 1, debe contener el listado de los alumnos y sus respectivas notas, un ejemplo de línea de cada archivo sería

Alumno01, nota1, nota2, nota3, nota4...

3. Implemente una función que facilite la lectura de un archivo de notas y almacene la data en una estructura de datos. Se recomienda el uso de diccionario
4. Genere un diccionario donde la clave sea el alumno y contenga las notas de cada materia.

Hint: Este diccionario se compone de clave nombre alumno y valor puede ser un diccionario con las asignaturas y la lista de notas, tal como se muestra a continuación:

```
{“Alumno 01”: {  
    “Inglés”: [1,2,3,4],  
    “Matemáticas”: [2,3,5,7],  
    “Ciencias”: [2,6,7]  
}}
```

5. Genere un método que obtenga el promedio de cada alumno por cada asignatura. Al igual que en el punto anterior, se recomienda que genere un diccionario con la nota obtenida.

6. Genere un método que determine si el alumno reprueba o no cada asignatura, considere para ello que la nota mínima en cada ramo es de un 4.0
7. Genere un método que obtenga el promedio del curso en cada asignatura.
8. Genere un método que obtenga los valores máximos y mínimos de cada asignatura.
9. Genere un método que obtenga la desviación estándar de cada asignatura.
10. Exporte la información del promedio de cada alumno, en un archivo personalizado por alumno, donde se muestre el nombre del alumno, las asignaturas y el promedio obtenido en cada una de ellas.
11. Exporte el resumen del rendimiento obtenido del curso en cada asignatura contemplando los promedios, máximos y mínimos, el alumno con mejor rendimiento y el alumno con peor rendimiento. Así como también el número de aprobados por cada asignatura.

***Hint: Se recomienda que los resúmenes por curso tengan el siguiente contenido en formato:***

Nombre Asignatura:

Cantidad de Alumnos:

Cantidad de Notas:

Promedio:

Desviación Estándar:

Nota Máxima:

Nota Mínima:

Mejor Alumno:

Peor Alumno:

Número de Aprobados:

Número de reprobados:

## **El algoritmo del ascensor**

El funcionamiento de un ascensor consiste en un algoritmo que dicta o decide al presionar un botón cuál será el que se empleará. De manera general se definen las siguientes reglas:

1. Dado el piso que en el que me encuentro, siempre el ascensor que me llevará a mi destino será el que esté más cercano.
2. Si existen dos o más ascensores con una misma distancia, se seleccionará el ascensor según el sentido, esto es, si subiré, se trae el ascensor que deba subir para llegar al destino. Importante, siempre se privilegia el que esté más cercano, independiente del sentido.
3. Siempre se debe consultar el estado del ascensor.

Dado lo expuesto previamente, se le solicita implementar el algoritmo del ascensor y hacer las simulaciones considerando un número de iteraciones variable. De manera adicional, es necesario

considerar el número de ascensores a trabajar y el estado de cada uno. Para ello, se espera que exista un archivo de configuración donde indique los siguientes puntos:

Ascensor Número:

Estado:

Piso actual:

Este archivo se debe actualizar a medida que se vaya generando las simulaciones, otro punto relevante, es que este script se ejecutará en paralelo, simulando 4 posibles ascensores, de tal manera que dicho archivo deberá estar siempre activo y será el que controle todos los estados.

Importante mencionar que al ejecutar el script, se debe mostrar el estado de los ascensores.

***Hint: Los 4 ascensores son simplemente 4 ejecuciones del script al mismo tiempo***