

# Taller 3

# Ejercicio 1.

---

Construya la clase **Computador** con atributos memoria (megabytes) y procesador (gigaflops). La memoria y el procesador no pueden ser manipulados desde fuera de la clase. Debe además crear el *constructor*.

Luego, cree una instancia de la clase **Computador**, asignándole 4096 megabytes de memoria y 5.43 gigaflops de procesador.

## Ejercicio 2

---

Construya la clase **Programa** con atributos consumoMemoria (megabytes) y cargaTrabajo (gigaflop); el consumo de memoria y la carga de trabajo no deben poder ser modificados desde el exterior de la clase. Debe además crear el *constructor* de la clase.

## Ejercicio 3

---

Construya la clase **SistemaOperativo** a partir de la clase **Programa**. La clase **SistemaOperativo** posee el atributo notebook de la clase **Computador**.

Inicialice notebook con valores al azar para la memoria (entre 4096 y 32768 megabytes) y el procesador (entre 0.25 y 10 gigaflops).

## Ejercicio 4

---

En la clase **SistemaOperativo**, implemente el método *ejecutar* que recibe como parámetro un **Programa**. Antes de ejecutar el programa el S0 debe comprobar si tiene memoria suficiente para ejecutarlo, mostrando cuánto tiempo (en segundos) le tomará ejecutar el programa. Considere en la operación, la carga de trabajo del S0, además de la del programa.

# Ejercicio 5

---

Cree una instancia de la clase **SistemaOperativo** y haga que ejecute un **Programa**.

El SO debe tener un consumo al azar de memoria (entre 1536 y 4096 megabytes) y carga de trabajo (entre 10 y 1000 gigaflop).

El programa debe tener un consumo de memoria y carga de trabajo asignado por el Usuario (por teclado).

## Ejercicio 6

---

Incluya en la clase **Computador** el atributo red, que representa la velocidad (en Mbps) a la cual puede transmitir archivos el computador.

Inicialice en la clase **SistemaOperativo** la velocidad de red con un valor al azar (entre 1 y 950 Mbps).

# Ejercicio 7

---

Incluya a nivel de **SistemaOperativo** el método *enviarArchivo* y envíe un archivo entre dos **Computadores**. Para *enviarArchivo* se requiere conocer el peso del archivo en megabytes y el Computador de destino; la transferencia se realizará a la menor velocidad de red disponible entre ambos **Computadores**. Indique el tiempo estimado de transferencia del archivo.



# Tarea

— — —

Diseñe la clase **Email** con atributos nombre, direccion, ultimoMensaje y contrasena. La contrasena no debe ser visible desde fuera de la clase, pero sí debería poder ser modificada. La direccion (ej: [mi@correo.cl](mailto:mi@correo.cl)) no puede ser modificada una vez creado el **Email**, pero si el nombre.

Implemente los métodos *enviar*, *recibir* y *leer*. El método *enviar* recibe como parámetro un mensaje y un **Email**. El método *recibir* guarda un mensaje recibido en el atributo ultimoMensaje. El método *leer* muestra el ultimoMensaje recibido.