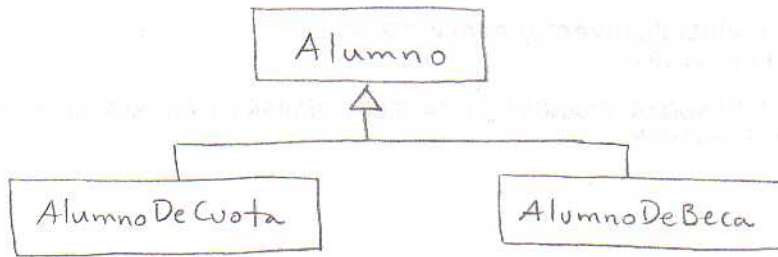


## Practica Individual

Desarrolle un programa en Java donde se implemente la siguiente relación jerárquica.



Las características de la superclase son:

Alumno
codigo : int nombre, carrera : String
+ Alumno (int codigo, String nombre, String carrera) // familia de metodos mostrar // familia de metodos cambiar // familia de metodos regresar + main (String args[]) : void

Las características exclusivas para c/u de las subclases son:

AlumnoDeCuota
cuotasemestral : double
+ AlumnoDeCuota (int codigo, String nombre, String carrera, double cuotasemestral) + mostrarCuotaSemestral () : void + cambiarCuotaSemestral (double cuotasemestral) : void + regresarCuotaSemestral () : double

AlumnoDeBeca
tipodebeca : String
+ AlumnoDeBeca (int codigo, String nombre, String carrera, String tipodebeca) + mostrarTipoDeBeca () : void + cambiarTipoDeBeca (String tipodebeca) : void + regresarTipoDeBeca () : String

En el main implementaremos los siguientes menus, además declarar los dos arreglos de referencias.

┌ \* Menú principal \* ┐

1. Altas
2. Bajas
3. Cambios
4. Consultas
5. Salir

└ Elige tu opción: ┘

┌ \* Menú Altas \* ┐

1. Agregar Alumno de Cuota
2. Agregar Alumno de Beca
3. Regresar al menú principal

└ Elige tu opción: ┘

┌ \* Menú Consultas \* ┐

1. Consultas individuales
2. Consultas generales para ADC
3. Consultas generales para ADB
4. Regresar al menú principal

└ Elige tu opción: ┘

AlumnoDeCuota adc[] = new AlumnoDeCuota[tamaño];  
AlumnoDeBeca adb[] = new AlumnoDeBeca[tamaño];

- Algoritmo Altas - (ADC o ADB)

pedir los datos

leer los datos

crear el objeto

incrementar contador

regresar al menú altas o al menú principal

⇓  
- Algoritmo Consultas Individuales -

pedir el código

leer el código

buscar al alumno en los dos arreglos

si se encuentra mostrar información del alumno

sino el alumno no existe

Nota: Para la opción consultas generales (ya sea ADC o ADB), se mostraran todos los alumnos ordenados por código.

┌ Dame el código del alumno a buscar: ┐

⇓



Dame el código del alumno a modificar:

- Algoritmo Cambios -

pedir el código

leer el código

buscar al alumno en los dos arreglos

si se encuentra ofrecer el menú apropiado, según el tipo de alumno que sea

sino el alumno no existe

Si el alumno encontrado es un alumno de cuota ofrecer el siguiente menú.

\* Menú Cambios para ADC \*

1. Codigo
2. Nombre
3. Carrera
4. Cuota semestral
5. Regresar al menú principal

Elige tu opción:

Si el alumno encontrado es un alumno de beca ofrecer el siguiente menú.

\* Menú Cambios para ADB \*

1. Código
2. Nombre
3. Carrera
4. Beca
5. Regresar al menú principal

Elige tu opción:

Nota: Para cualquiera de las opciones (1 a 4) solicitar el nuevo dato y llevar a cabo la modificación correspondiente.

[ Dame el código del alumno a eliminar: ]

### - Algoritmo Bajas -

pedir el código

leer el código

buscar el alumno en los dos arreglos

si se encuentra eliminarlo y decrementar contador

sino el alumno no existe

### Notas Adicionales:

• Validar al menos 5 aspectos importantes. Ejemplos de algunos aspectos relevantes.

a) Que los códigos no se repitan

b) Que cuando los arreglos se llenen ya no se pueda dar de alta más alumnos

c) Que si no se ha dado de alta ningún alumno y el usuario solicita alguna opción del menú consultas se muestren los o el mensaje(s) apropiado(s).

d) Que cuando el programa solicite un entero y el usuario por error teclee una cadena, el programa no truenre.

e) Que si el usuario introduce una opción no válida en algún menú, el programa muestre el mensaje apropiado y continúe trabajando.

Nota: Utilizar al menos un bloque try-catch en la parte de las validaciones.

### Evaluación - Práctica Individual

Altas 5 pto.

Bajas 5 pto.

Cambios 5 pto. (2.5 pto. c/u)

Consultas I 5 pto.

Consultas G 5 pto. (2.5 pto. c/u)

Validaciones 5 pto. (1 pto. c/u)

30 pto.