

## Repàs bàsic JAVA

### Exercicis proposats: E/S i operacions bàsiques:

**Crea una classe per a cada exercici i implementa el mètode principal ()**).

1. Programa que llegeix el tipus de conductor i el percentatge d'alcohol en sang i li retorna si vostè està actuar o no per conduir.

Donat:

**Tipus 1:** La alcoholpermès és **0,3** per als conductors de càrrega, vehicles especials (per exemple, autobús, taxi, ambulància...) i novells.

**Tipus 2:** La taxa d'alcohol per a conductors i ciclistes normals és de **0,5**.

2. Programa que llegeix per teclat el valor dels tres costats d'un angle tri i ha de ser comprovar si aquests tres costats formen o no un triangle. Tres costats defineixen un triangle si la suma de la longitud de qualsevol dos costats és més gran que la longitud del tercer costat.

### Instruccions de control.

Si utilitzeu algun mètode diferent del principal() serà dins de la mateixa classe i serà estàtic.

3. Programa que demana a l'usuari un número i calcula tots els divisors menors d'aquest mateix número, mostrant-los a la pantalla.
4. Un programa que llegeix un enter i positiu i escriu tres columnes. El primer compta d'un a un al nombre escrit comptant-ne un a la vegada; la segona columna compta de dos per dos i la tercera de tres de cada tres

### Matrius:

El vector o matriu utilitzada serà un atribut de la classe. Utilitzeu els mètodes que vostè pensa que és convenient per resoldre cada problema

5. Realitzeu un programa que ompli un vector de 15 posicions amb números generats aleatòriament (entre 1 i 90) i el presenti a la pantalla. Els 15 enters representen un hipotètic cartró del joc bingo. Recordeu, els números no es poden repetir.

6. Creeu un programa **anomenat venedors** que creï una matriu de 18 X 10 que indiqui que som propietaris d'una empresa de 18 proveïdors cadascun dels quals ven 10 productes.  
La matriu emmagatzema els ingressos obtinguts per cada venedor en cada producte, de manera que un menú li permet emmagatzemar els ingressos, revisar el total de cada venedor i obtenir els ingressos totals.

## Col·leccions:

7. Implementar un programa de gestió de classes sense assumir cap nombre màxim d'estudiants. S'utilitzarà una llista per a aquest propòsit.

Per a cada estudiant, s'emmagatzemarà la següent informació: Nom, Cognoms, DI, Nota al número i Nota a la lletra. Crea una llum de classe **AA..**

Implementa la classe **del curs** l'atribut de la qual és una llista del tipus **Estudiant**.

El programa permetrà les següents operacions, en un menú d'opcions:

- Afegeix un estudiant nou.
- Suprimiu l'estudiant existent (donant la seva posició a la llista i/o el seu nom, segons el que preferiu).
- Canvia les dades dels estudiants: Modifica els teusalumnes...
- Mostra totes les dades de tots els estudiants (un per un, per exemple).
- Canvieu la nota de lletra de cada estudiant segons la seva nota numèrica: Not Presented (<0), Suspens (<5), Approved (<7), Notable (<9), Outstanding (<10) i Honors (10).
- Mostrar el nombre total d'estudiants de cada nota (Suspens, Aprovat...) i el percentatge que assumeixen respecte al nombre total d'estudiants de la llista i respecte al nombre total d'estudiants presentats (excloent-ne l'UNRE presentat).

8. Implementar un programa per gestionar una guia telefònica sense assumir cap nombre màxim de contactes. S'utilitzarà una llista per a aquest propòsit.

Per a cada contacte s'emmagatzemarà la informació següent: Nom i Telèfon al número. Crea la classe Contactes. **Contactos**

Implementa la classe **Calendari** l'atribut de la qual és una llista del tipus **Contactes**.

El programa permetrà les següents operacions, en un menú d'opcions:

- Afegeix contacte en ordre alfabètic.
- Cerqueu contacte.
- Suprimeix el contacte
- Ensenya-ho tot.
- Canvia el número de telèfon.

## Diverses classes

9. Crea un programa que contingui els elements següents:

- Una classe **de punts** que representa punts que consisteixen en les seves coordenades x i y.
- Una classe **de triangle** que representa triangles utilitzant tres vèrtexs (cada vèrtex és un element de la classe punt) punto)
- Un mètode anomenat **mostraCoordenades()** que mostra les coordenades dels punts en aquest format: (2,3) (la x és dues i la y és tres)
- Un mètode anomenat **canviaCoordenades()** que permet canviar les coordenades del joc de paraules al qual s'envia com a paràmetre.
- Un mètode de trucada **distancePoints()** que retorna la distància entre dos punts (que rep com a paràmetres del mètode)
- Una funció anomenada **Triangle d'escriptura** que escriu els vèrtexs del triangle en aquesta forma, per exemple:  
(3,2),(8,3),(7,5)
- Una funció anomenada **perimeterTriangle** que permet escriure el perímetre d'un triangle determinat. Per arreglar-ho, podem entendre que el perímetre d'un triangle és la suma de les distàncies entre cada vèrtex
- Un mètode principal() que prova els mètodes.

#### 10. Crea un programa que contingui els elements següents:

- Una classe anomenada **Titular** que representa els clients amb el seu nom, cognoms, adreça i id
- Una classe anomenada **Compte Corrent** que té com a camps, dos titulars, un número de compte (serà de 20 caràcters) i el saldo del compte (*double*)
- Creeu l'emplenament **FunctionAccount** que retorna com a resultat un compte les dades del qual han de ser llegides per teclat, però tenint en compte que: Pot haver-hi un o dos títols (almenys un), el número de compte ha de ser **exactament** de 20 números i el saldo ha de ser positiu
- Creeu el **canvi FunctionSaldo** que rep un compte i un valor decimal i canvia el saldo del compte perquè coincideixi amb el valor del nombre decimal