

## **AP2. Implementació en Java**

*Alejandro Garcia Dopico*

### **Milestones (Fites i objectius)**

En finalitzar aquesta activitat, l'alumne ha d'assolir i ser competent en les següents fites:

1. Analitza i dissenya els possibles algorismes per resoldre problemes.
2. Crea projectes de desenvolupament d'aplicacions i utilitza entorns integrats de desenvolupament.
3. Identifica els diferents tipus de variables i la utilitat específica de cadascun.
4. Modifica el codi d'un programa per crear i utilitzar variables.
5. Crea i utilitza constants i literals.
6. Classifica, reconeix i utilitza en expressions els operadors del llenguatge.
7. Comprova el funcionament de les conversions de tipus explícites i implícites.
8. Introdueix comentaris en el codi.
9. Escriu i prova programes senzills reconeixent i aplicant els fonaments de la programació.
10. Utilitza estructures de dades simples i compostes.
11. Escriu i prova codi que faci ús de les estructures de selecció.
12. Utilitza correctament les diferents estructures de repetició disponibles.
13. Reconeix les possibilitats de les sentències de salt.
14. Realitza operacions bàsiques, compostes i de tractament de caràcters.
15. Revisa i corregeix els errors apareguts en els programes.
16. Comenta i documenta adequadament els programes realitzats.

### **Desenvolupament**

Llegeix atentament l'enunciat de cada activitat. Cal llegir atentament cada exercici i proposar una solució a cada exercici. Es pot consultar qualsevol font d'informació (s'ha d'indicar la font dins l'apartat bibliografia) però s'ha de proposar la solució amb les pròpies paraules (no es pot presentar una còpia literal o fragments).

### **Entrega**

L'exercici s'ha de pujar al repositori personal de GitHub abans del termini indicat a la tasca (codi font dels programes).

Cal entregar un document en format .pdf amb la graella d'autoavaluació per a cada exercici.



### **Enunciat**

1. Implementa els exercicis [1-9] (inclosos) del següent [document](#) (apartat "Implementació en Java").

*Les activitats estan pujades al github amb el nom "UF1/AP2/AP2\_Activitats"*

2. Escull 3 exercicis de 3 companys diferents (1 company per exercici) que creguis que es pot millorar/optimitzar. Copia'l i modifica'l, indicant els canvis/millores i qui és l'autor original en els comentaris. Fes un petit video per a cada exercici utilitzant l'eina debug i explica com funciona en cada codi (amb la teva veu). Afegeix l'enllaç al vídeo com a comentari dins del codi font.

*Les activitats estan pujades al github amb el nom "UF1/AP2/AP2\_Videos"*

3. Puja al teu repositori tots els exercicis (pots crear una única carpeta amb tot els source files o un projecte per a cada exercici, dins d'una carpeta contenidora).



### Avaluació

Per a cada exercici, omple la graella següent:

Exercici X					
Criteri	2 - Molt bé	1 - Suficient	0 - Insuficient	Dedicació (minuts)	Punts
<b>Robust</b>	El programa s'executa sense problemes i no mostra errors en execució	El programa s'executa, però en algun punt de l'execució falla i finalitza o mostra errors en execució	El programa no s'executa (dóna errors en compilació/execució)	<b>Comentari a baix</b>	<b>2</b>
<b>Claredat i ordre</b>	Es pot seguir fàcilment la seqüència de les instruccions. És endreçat i conserva l'estructura dels exemples vistos a classe i als materials.	El codi no està correctament estructurat: les estructures condicionals no discriminen casos sinó que repeteixen sovint comprovacions o el flux d'execució és més complex del necessari. A trossos les indentacions no són correctes.	No segueix en absolut l'estructura dels exemples. L'algoritme (idea que hi ha darrera del programa) és desendreçat. No està gens indentat.		<b>2</b>
<b>Variables</b>	Els noms de les variables són prou autodescriptius. Els tipus de dades utilitzats són els adequats (els més eficients)	Els noms de les variables són autodescriptius, però no tots els tipus de dades són correctes. No totes les variables estan declarades a	Els noms de les variables no són autodescriptius (no ens informen del seu contingut). Algun dels tipus no són correctes. Les variables no es declaren al principi		<b>2</b>



	per a la tasca requerida). Les variables es declaren a principi del codi.	principi del codi.	del codi (es poden trobar declaracions al llarg del programa).		
<b>Constants</b>	Totes les dades que tenen un valor que no varia en tot el codi estan definides com a constants. Els noms de les constants és en MAJÚSCULES	Defineix com a constant algunes de les dades, però no d'altres que serveixen per marcar valors mínims i màxims i que no canvien en tot el codi.	No defineix constants. Els noms de les constants no estan en MAJÚSCULES.		<b>2</b>
<b>Bucles</b>	Fa servir correctament els bucles, tal i requereix l'algorisme per a executar-se de la manera més eficient.	Fa servir els bucles però no controla correctament els valors límits, no incrementa correctament el comptador o la condició de sortida és incorrecta/inexistent..	No utilitza els bucles correctament.		<b>2</b>
<b>Correcte</b>	Fa tot el que es demana. Passa correctament totes proves del joc de proves (debug).	Fa una part de les coses que es demanen, però no totes.	No fa el que es demana.		<b>2</b>

## Comentari

La duració de tots els exercicis a sigut d'unes 8 hores, tots els exercicis funcionen. Ha sigut fàcil perquè la lògica del pseudocodi era la mateixa i m'ha facilitat també buscar per internet com fer algunes coses, sobretot utilitza'n aquest enllaç;  
<https://www.w3schools.com/java/>