



UT2
José R. Mas Davó authored 1 month ago

70e0d2e8

M+ UT2Problema5.md 8.12 KiB

UT2 - Problema 5: Estadística de notes

Agrupament: Individual



Pregunta guia

Com podem desenvolupar un programa que ajude un professor a analitzar estadísticament els resultats d'un examen, processant un nombre indefinit de notes i proporcionant informació detallada sobre el rendiment dels alumnes?

Objectius d'aprenentatge

En completar aquesta pràctica, sereu capaços de:

- Escriure i provar codi que faça ús d'estructures de selecció i control.
- Utilitzar estructures de repetició per processar un nombre indefinit d'entrades.
- Gestionar i evitar errors comuns com la divisió entre zero i entrades de dades no vàlides.
- Realitzar càlculs estadístics bàsics (mitjanes, percentatges, valors màxims i mínims).
- Formatar i presentar resultats estadístics de manera clara i comprensible.
- Provar i depurar els programes.
- Comentar i documentar el codi.

Context i descripció del problema

Un professor de programació en Java necessita una eina per analitzar ràpidament els resultats dels exàmens finals del seu curs. Vol obtenir una visió global del rendiment dels seus alumnes, incloent-hi percentatges d'aprovat i suspesos, notes mitjanes i valors extrems.

L'objectiu és elaborar un programa que permeti al professor introduir les notes dels alumnes com a valors numèrics que poden contenir decimals. Es podran introduir les notes que es vulga fins que s'introduïska un valor negatiu a la nota, la qual cosa indicarà el final de l'entrada de dades.

Una volta finalitzada la introducció de notes, el programa ha de ser capaç de mostrar:

1. El nombre d'alumnes avaluats.
2. El nombre d'alumnes aprovats.
3. El percentatge d'alumnes aprovats sobre el total.
4. El nombre d'alumnes suspesos.
5. El percentatge d'alumnes suspesos sobre el total.

- 6. La nota més alta.
- 7. La nota més baixa.
- 8. La nota mitjana dels alumnes en general.
- 9. La nota mitjana dels alumnes aprovats.
- 10. La nota mitjana dels alumnes suspesos.

El problema a resoldre

Desenvolupar un programa que reculli i analitzi estadísticament les notes dels alumnes, permetent la introducció de dades fins que s'introduísca un valor negatiu. El programa ha de calcular i mostrar diverses estadístiques sobre els resultats dels alumnes, controlant els possibles errors que es puguin produir a l'hora d'introduir les notes per teclat, de manera que si s'introdueix un valor invàlid, es sol·licitarà que es torne a introduir per teclat. També s'ha de controlar que no es produïsqen errors de divisió entre zero.

Què necessites per a fer aquesta activitat?

- Ordinador amb un editor de text i el compilador "javac" funcionant (Pots utilitzar NetBeans si ho prefereixes).

Requisits del programa

1. Entrada de dades:

- Demanar a l'usuari les notes dels alumnes com a valors numèrics que poden contenir decimals.
- Continuar demanant notes fins que s'introduísca un valor negatiu.
- Validar que les notes estiguen dins del rang vàlid (0-10).

2. Processament:

- Calcular el nombre d'alumnes avaluats.
- Calcular el nombre i percentatge d'alumnes aprovats (nota ≥ 5.0).
- Calcular el nombre i percentatge d'alumnes suspesos (nota < 5.0).
- Determinar la nota més alta i la nota més baixa.
- Calcular la nota mitjana general, la nota mitjana dels aprovats i la nota mitjana dels suspesos.
- Assegurar que les dades introduïdes són vàlides i evitar errors com la divisió entre zero.

3. Eixida de dades:

- Mostrar les estadístiques calculades després de la introducció de les notes.
- Formatar adequadament els percentatges i les notes mitjanes (amb dos decimals).
- Mostrar un missatge d'error si s'introdueixen dades incorrectes i sol·licitar una nova entrada.

Exemple de funcionament del programa:

```
-----
--  ESTADÍSTICA DE NOTES  --
-----
```

```
Introdueix una nota (valor negatiu per a finalitzar): 6.2
Introdueix una nota (valor negatiu per a finalitzar): 3.1
Introdueix una nota (valor negatiu per a finalitzar): 12.5
ERROR: Nota invàlida
Introdueix una nota (valor negatiu per a finalitzar): 9.7
Introdueix una nota (valor negatiu per a finalitzar): 0.8
Introdueix una nota (valor negatiu per a finalitzar): 2A
ERROR: Nota invàlida
Introdueix una nota (valor negatiu per a finalitzar): 5.3
Introdueix una nota (valor negatiu per a finalitzar): -8
```

```
- S'han avaluat un total de 5 alumnes
- La nota mitjana del curs ha sigut un 5,02
- Han aprovat 3 alumnes, el que suposa un 60,00% del total
- La nota mitjana dels alumnes aprovats ha sigut un 7,07
- Han suspés 2 alumnes, el que suposa un 40,00% del total
- La nota mitjana dels alumnes suspesos ha sigut un 1,95
- La nota més alta ha sigut un 9,70
- La nota més baixa ha sigut un 0,80
```

```
-----
```

```
--  ESTADÍSTICA DE NOTES  --
-----

Introdueix una nota (valor negatiu per a finalitzar): 15
ERROR: Nota invàlida
Introdueix una nota (valor negatiu per a finalitzar): AZ
ERROR: Nota invàlida
Introdueix una nota (valor negatiu per a finalitzar): -1

No s'han introduït notes.
```

```
-----
--  ESTADÍSTICA DE NOTES  --
-----

Introdueix una nota (valor negatiu per a finalitzar): 2.1
Introdueix una nota (valor negatiu per a finalitzar): 4.3
Introdueix una nota (valor negatiu per a finalitzar): 0
Introdueix una nota (valor negatiu per a finalitzar): -2

- S'han avaluat un total de 3 alumnes
- La nota mitjana del curs ha sigut un 2,13
- Han aprovat 0 alumnes, el que suposa un 0,00% del total
- La nota mitjana dels alumnes aprovats és inexistent, perquè no hi ha cap aprovat
- Han suspés 3 alumnes, el que suposa un 100,00% del total
- La nota mitjana dels alumnes suspesos ha sigut un 2,13
- La nota més alta ha sigut un 4,30
- La nota més baixa ha sigut un 0,00
```

Instruccions

1. Investigació preliminar:

- Repassa com manejar entrades de dades amb valors decimals en Java.
- Repassa l'ús d'estructures de control i repetició per processar entrades múltiples.

2. Desenvolupament del programa:

- Implementa una estructura repetitiva per a llegir les notes fins que s'introduísca un valor negatiu.
- Crea variables per emmagatzemar les dades necessàries (suma de notes, comptadors, etc.).
- Implementa la lògica per calcular totes les estadístiques requerides.
- Afig validació d'entrada per assegurar que les notes estiguen dins del rang vàlid.
- Utilitza estructures condicionals per evitar errors de divisió per zero.
- Afig comentaris al codi per explicar cada pas i facilitar la comprensió.

3. Proves i depuració:

- Prova el programa amb diferents conjunts de notes, incloent-hi:
 - Notes variades dins del rang vàlid.
 - Intents d'introduir notes fora del rang vàlid o valors invàlids.
 - Un conjunt de notes on tots els alumnes aproven o suspenen.
- Verifica que totes les estadístiques es calculen correctament.
- Comprova que el format de presentació siga clar i consistent.

4. Reflexió i documentació:

- Escriu un informe breu que incloga:
 - Una explicació de com has implementat el processament de les notes i el càlcul de les estadístiques.
 - Com has gestionat la validació d'entrada i els possibles errors.
 - Els reptes que has enfrontat durant el desenvolupament i com els has superat.
 - Suggestiments per a futures millores o funcionalitats addicionals.

Entrega

- Envia el fitxer del codi font (`UT2Problema5.java`) i l'informe en un document PDF.
- Recorda seguir la convenció de noms.
- **Important:** Afig el teu nom i cognoms sempre com a comentari al principi dels fitxers JAVA (En NetBeans després de `@author`).

Extensió opcional

Per als estudiants que acaben abans o vulguen un repte addicional:

- Implementa la possibilitat de categoritzar les notes en diferents rangs (excel·lent, notable, bé, suficient, insuficient) i mostra estadístiques per a cada categoria.