Update ProjecteEndevinar-EstructuresControl.md
José R. Mas Davó authored 3 hours ago

e6075ce0

ProjecteEndevinar-EstructuresControl.md 6.18 KiB

Endevina la Paraula - 2. Estructures de control

1. Introducció

1.1. Context i justificació

En aquesta segona fase del projecte "Joc d'Endevinar la Paraula", els estudiants ampliaran la funcionalitat del programa desenvolupat en la primera fase. Aquesta fase se centrarà en l'aprenentatge i aplicació de les estructures de control, com ara condicionals i bucles. Això permetrà implementar múltiples intents, validació d'entrada i lògica de joc més complexa.

1.2. Objectius generals d'aquesta fase

- Implementar estructures de control (if, else, switch, while, do-while, for).
- Millorar la interacció amb l'usuari permetent múltiples intents.
- Introduir validació d'entrada i gestió d'errors bàsica.
- Ampliar la lògica del joc amb pistes i puntuació.

1.3. Competències a desenvolupar

- Capacitat per utilitzar estructures de control per gestionar el flux del programa.
- Habilitat per crear bucles per a la repetició de tasques.
- Capacitat per validar condicions en els intents de l'usuari.

1.4. Pregunta guia

Com podem utilitzar les estructures de control per millorar la jugabilitat i la robustesa del nostre joc d'endevinar paraules?

2. Descripció del repte

2.1. Situació inicial

Partim del joc bàsic desenvolupat en la fase anterior, que permet un sol intent d'endevinar una paraula predefinida.

2.2. Necessitats a cobrir

- Implementar múltiples intents per endevinar la paraula.
- Afegir un sistema de puntuació basat en el nombre d'intents.
- Proporcionar pistes després de cada intent fallit.
- Validar l'entrada de l'usuari per assegurar que és una paraula vàlida.
- Permetre a l'usuari triar entre continuar jugant o sortir del joc.

2.3. Restriccions i consideracions

- Limitar el nombre màxim d'intents (per exemple, a 6).
- La paraula a endevinar seguirà sent predefinida en aquesta fase.
- El programa ha de ser clarament estructurat i fàcil d'entendre.
- S'ha de proporcionar feedback a l'usuari sobre els intents realitzats.

3. Planificació i organització

3.1. Formació d'equips

Els alumnes seguiran treballant individualment en aquesta fase.

3.2. Cronograma i fites

- Revisió i correcció del codi de la primera fase i planificació de les tasques.
- Disseny de l'estructura del programa ampliat.
- Implementació de bucles per a múltiples intents.

- Desenvolupament del sistema de puntuació i pistes.
- Implementació de la validació d'entrada i gestió d'errors.
- Proves, depuració i refinament.

3.3. Recursos necessaris

- IDE Java (preferiblement NetBeans)
- Documentació de Java sobre estructures de control (condicionals i repetitives)
- Materials de suport sobre algorismes de comparació de cadenes

4. Fase d'investigació

4.1. Identificació de coneixements previs

Revisió dels conceptes d'estructures de control: if-else, switch, while, do-while, for.

4.2. Fonts d'informació a consultar

- Apunts de classe sobre estructures de control.
- Documentació oficial de Java sobre estructures de control.
- Tutorials en línia sobre implementació de jocs simples en consola.

5. Desenvolupament del projecte

5.1. Tasques específiques a realitzar

- Revisar i corregir el codi desenvolupat en la primera fase.
- Implementar un bucle principal per permetre múltiples partides.
- Crear un bucle per gestionar els intents dins de cada partida.
- Afegir un sistema de pistes que indique si la paraula introduïda és més llarga, més curta o igual a la paraula secreta.
- Implementar un sistema de puntuació.
- Afegir validació d'entrada per assegurar que l'usuari introdueix una paraula vàlida.

5.2. Aplicació de coneixements i habilitats

Utilitzar les estructures de control per millorar la lògica i la interactivitat del joc.

5.3. Creació del producte o solució

El programa ha de:

1. Mostrar un menú principal amb opcions per jugar o eixir.
2. En cada partida, permetre múltiples intents per endevinar la paraula (podria elegir-ho el propi jugador).
3. Proporcionar pistes després de cada intent fallit.
4. Mostrar la puntuació final basada en el nombre d'intents.
5. Preguntar si l'usuari vol jugar de nou o eixir.

5.4. Revisió i millora contínua

Proves exhaustives de totes les rutes possibles del programa i refinament del codi.

6. Criteris d'avaluació

6.1. Rúbrica d'avaluació

Es valorarà:

- Correcta implementació de les estructures de control.
- Lògica adequada per a la gestió de múltiples intents i partides.
- Implementació efectiva del sistema de pistes i puntuació.
- Robustesa en la validació d'entrada i gestió d'errors.
- Claredat i eficiència del codi.

6.2. Entregables esperats

- Codi font complet del projecte ampliat.
- Diagrama de flux que represente la lògica del joc.
- Breu informe explicant les millores implementades i les decisions de disseny.

6.3. Ponderació de cada part

- Convenció de noms (5%)
- Execució (10%)
- Implementació correcta de les estructures de control (20%)
- Validació d'entrada i gestió d'errors (20%)
- Funcionalitat del joc (múltiples intents, pistes, puntuació) (25%)
- Qualitat del codi i documentació (10%)
- Presentació al final del projecte (10%)

7. Presentació

En esta segona fase no hi haurà presentació.

8. Reflexió i metacognició

8.1. Preguntes per a l'autoavaluació

Com han millorat les estructures de control la funcionalitat del nostre joc? Quins reptes hem trobat en la implementació de múltiples intents i la gestió d'errors?

8.2. Anàlisi del procés d'aprenentatge

Reflexió sobre com l'ús d'estructures de control ha canviat la manera d'abordar el disseny del programa.

8.3. Proposta de millores per a futurs projectes

Idees per ampliar encara més el joc en futures fases, com ara implementar un sistema per generar paraules aleatòries o afegir nivells de dificultat.

Notes d'ajuda

1. Pots utilitzar `Math.max(a,b)` per obtindre el màxim entre dos valors.

   ```
   System.out.println("Puntuació: " + Math.max(0, puntsUsuari));
   ```

2. Per a obtindre la longitud de una paraula existeix el mètode `length()` dels `String`.

   ```
   String paraula = "Projecte";
   System.out.println("Lletres: " + paraula.length())
   ```

   Mostra per pantalla:

   ```
   Lletres: 8
   ```