TOKAJ-HEGYALJA EGYETEM

Comenius Intézet

**Magasszintű programozási nyelvek II. (THE\_00745\_N\_5\_G)**

**Akasztófa Beadandó feladat**

Konzulens:

Perlaki Attila

Tanár Úr

Készítette:

Szalai Dominik Programtervező informatikus-szak Nappali tagozatos

Sárospatak 2024

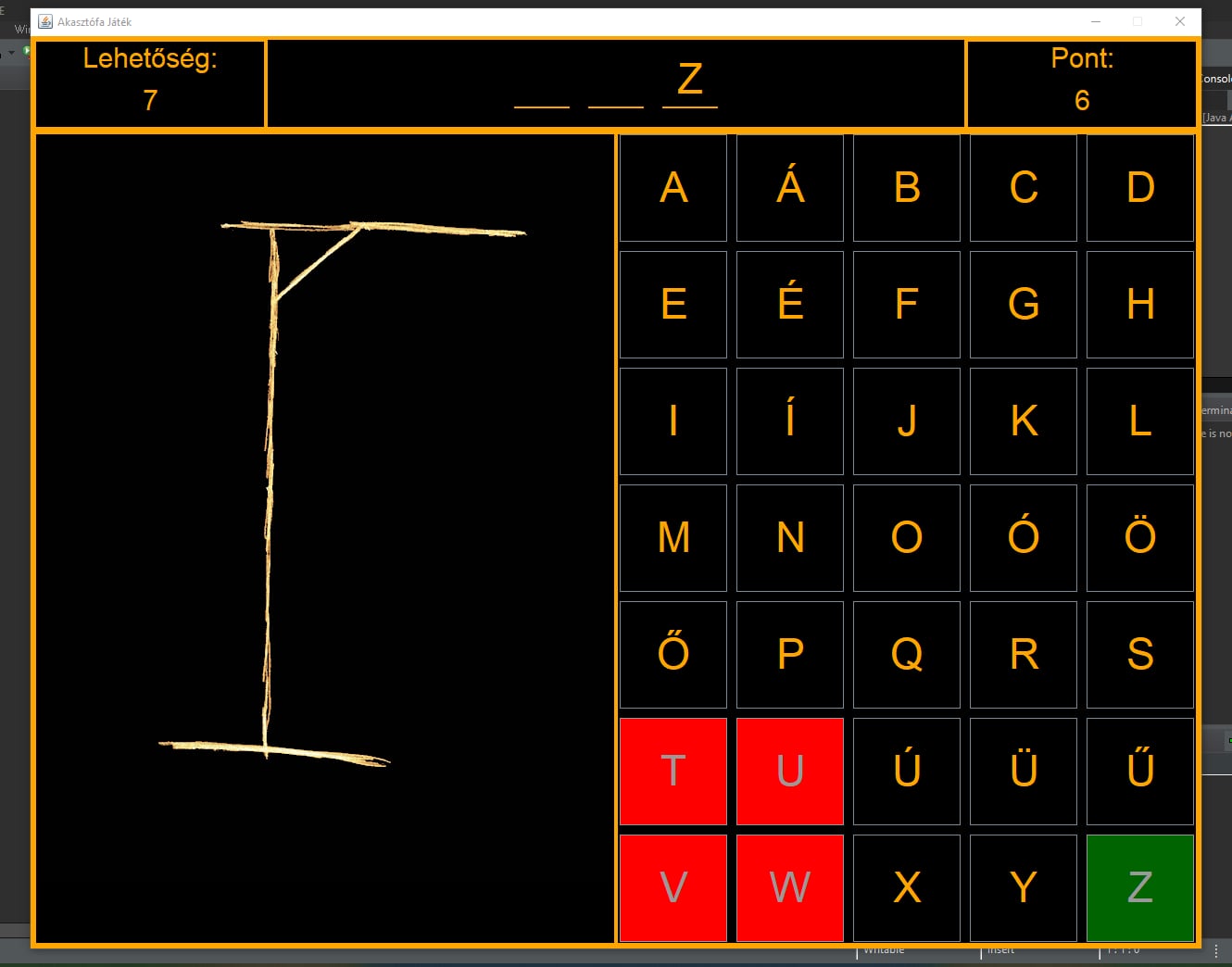
**Játék Leírás:**

A játék célja, hogy minél random kiválasztót szót találjunk ki, minél többet találunk ki annál nagyobb lesz a pontszámunk, ha elrontottuk a tippet többször akkor veszítünk és felírhatjuk magunkat a ranglistára

A játék Java nyelven irt

A ranglista SQLlite-al irt

Forráskód: <https://github.com/SzalDom/Akasztofa>



**Tartalom**

[1. Main 4](#_Toc166783796)

[2. Ablak 5](#_Toc166783797)

[3. JatekAblak 6](#_Toc166783798)

[4. MenuAblak 7](#_Toc166783799)

[5. Gombsor 8](#_Toc166783800)

[6. MegfejtesPanel 9](#_Toc166783801)

[7. Dialogus 10](#_Toc166783802)

[8. Beviteli 11](#_Toc166783803)

[9. Kerdes 12](#_Toc166783804)

[10. Adatbazis 13](#_Toc166783805)

[11. HighScoresPanel 14](#_Toc166783806)

[12. Szotar 15](#_Toc166783807)

[13. AdatbazisBejegyzes 16](#_Toc166783808)

[14. Pontrendszer 17](#_Toc166783809)

# Main

Funkció: A program belépési pontja. Inicializálja a menüablakot és az eseménykezelőt.

Fontosabb Metódusok:

main(String[] args): Inicializálja a MenuAblak objektumot és az EsemenyKezelo interfészt, amely az eseményeket

# Ablak

Funkció: Az alap ablak (JFrame) beállításait kezeli, amely minden ablak alapja.Fontosabb Metódusok:Ablak(): Beállítja az alapvető ablak tulajdonságokat, mint például méret, cím, elhelyezkedés és kilépési esemény.beallitSzineket(Container container): Rekurzív metódus, amely beállítja az ablak és annak komponenseinek színeit és stílusát.

# JatekAblak

Funkció: Az aktuális játékablak kezelése, ahol maga a játék folyik.

Fontosabb Metódusok:

JatekAblak(): Beállítja a játékablakot, inicializálja a képeket, betűket, és a játékhoz szükséges paneleket.

inicializal(): Inicializálja a játék állapotát újraindításakor vagy új játék kezdetekor.

rossz\_probalkozas(): Kezeli a helytelen próbálkozásokat és frissíti a próbálkozások számát.

frissitAbrat(int allapot): Frissíti a játék állapotát megjelenítő ábrát.

# MenuAblak

Funkció: A játék főmenüjét kezeli, ahol a játékosok elindíthatják a játékot vagy megtekinthetik a pontszámokat.

Fontosabb Metódusok:

MenuAblak(): Inicializálja a menü ablakot, hozzáadja a címeket, gombokat és a pontszámokat tartalmazó panelt.

# Gombsor

Funkció: A betűk gombjait kezeli, amelyeket a játékosok használhatnak a szavak kitalálására.

Fontosabb Metódusok:

Gombsor(): Beállítja a gombok elrendezését és létrehozza azokat.

createGombok(): Létrehozza a játékhoz szükséges betű gombokat és eseménykezelőket állít be.

gombotNyomtak(char betu): Absztrakt metódus, amely kezeli a gombok nyomását.

tilt(boolean tiltva): Beállítja, hogy a gombok aktívak vagy inaktívak legyenek.

helyreallit(): Visszaállítja a gombokat az alapértelmezett állapotba.

# MegfejtesPanel

Funkció: A játékos által kitalált betűk megjelenítését kezeli.

Fontosabb Metódusok:

MegfejtesPanel(): Inicializálja a panelt.

inicializal(String szo): Beállítja a kitalálandó szót és annak betűmezőit.

talalt(char betu): Frissíti a panelt, ha egy betűt helyesen találtak ki.

felfed(boolean sikerult): Felfedi a kitalálandó szót, ha a játék véget ért.

# Dialogus

Funkció: Általános dialógus ablak kezelése, amely visszajelzéseket ad a játékosnak.

Fontosabb Metódusok:

Dialogus(String cim, String szoveg, String gombNeve): Létrehozza a dialógus ablakot a megadott paraméterekkel.

kattintottak(String gomb\_neve): Absztrakt metódus, amely a gomb nyomásának kezelésére szolgál.

# Beviteli

Funkció: Játékos név bevitelét kezeli a pontok mentése előtt.

Fontosabb Metódusok:

Beviteli(Frame parent, String cim, int pont): Létrehozza a beviteli dialógus ablakot és kezeli a név mentését.

# Kerdes

Funkció: Kérdést tartalmazó dialógus ablak kezelése.

Fontosabb Metódusok:

Kerdes(String cim, String szoveg, String[] gombNevek): Létrehozza a kérdést tartalmazó dialógus ablakot.

kattintottak(String gomb\_neve): Absztrakt metódus, amely a gomb nyomásának kezelésére szolgál.

# Adatbazis

Funkció: Az adatbázis kezelését végzi, beleértve a pontszámok tárolását és lekérdezését.

Fontosabb Metódusok:

inicializal(): Inicializálja az adatbázist.

kerPontok(): Lekérdezi a pontszámokat az adatbázisból.

beszur(String username, int score): Új pontszámot szúr be az adatbázisba.

torolMaradekot(): Törli az adatbázisból a nem top pontszámokat.

# HighScoresPanel

Funkció: Megjeleníti a legmagasabb pontszámokat tartalmazó táblázatot.

Fontosabb Metódusok:

HighScoresPanel(): Inicializálja a táblázatot.

frissit(): Frissíti a táblázatot az adatbázisból lekért pontszámok alapján.

# Szotar

Funkció: A kitalálandó szavakat tárolja és kezeli.

Fontosabb Metódusok:

inicializal(): Inicializálja a szótárat a fájlból.

kerVeletlen(): Véletlenszerű szót ad vissza a szótárból.

# AdatbazisBejegyzes

Funkció: Az adatbázis pontszám bejegyzéseit reprezentálja.

Fontosabb Metódusok:

AdatbazisBejegyzes(String username, int score, String timestamp): Létrehozza a pontszám bejegyzést.

getUsername(), getScore(), getTimestamp(): Visszaadja a bejegyzés adatait.

# Pontrendszer

Funkció: Pontszámítást végzi a játék során.

Fontosabb Metódusok:

szamol(boolean jo\_valasz, char betu, int hatralevo\_probalkozasok): Kiszámolja a pontszámot a válasz alapján.

inicializal(): Inicializálja a pontszámítás rendszerét.

Következtetés

A fenti osztályok és azok metódusai együttesen alkotják az Akasztófa játékot, amely egy jól strukturált, objektumorientált program. Az osztályok közötti együttműködés és az eseményvezérelt programozás biztosítja a játék megfelelő működését és a felhasználói élményt. A részletes bemutatás segít megérteni a program felépítését és működését, valamint támogatást nyújt a további fejlesztésekhez és bővítésekhez.

Pontrendszer Működése

Pontszámítás

Helyes válasz esetén:

A szamol metódus meghívásakor, ha a jo\_valasz paraméter értéke igaz, a metódus ellenőrzi, hogy a kitalált betű szerepel-e a letterPoints térképen (Map).

Ha a betű megtalálható a térképen, visszaadja a betűhöz tartozó pontértéket.

Ha a betű nem található a térképen (azaz a betű a ritkább betűk közé tartozik), visszaadja a legritkabb változó értékét, ami 15 pont.

Helytelen válasz esetén:

A szamol metódus meghívásakor, ha a jo\_valasz paraméter értéke hamis, a metódus a hatralevo\_probalkozasok paraméter alapján határozza meg a levonandó pontokat.

Ha több mint 7 próbálkozás maradt, a metódus -1 pontot von le.

Ha 5 és 7 közötti próbálkozás maradt, a metódus -2 pontot von le.

Ha 4 vagy kevesebb próbálkozás maradt, a metódus -3 pontot von le.

Betűk Rangsorolása

A pontszámításhoz a betűk ritkaságuk alapján három kategóriába sorolhatók:

Legritkább betűk: 15 pontot érnek, ha helyesen találják ki őket. Ezek a betűk nincsenek explicit módon a letterPoints térképen tárolva, mert a getOrDefault metódus használatával alapértelmezett értékként 15 pontot kapnak, ha nem találhatók a térképen.

Közepesen ritka betűk: 10 pontot érnek. Ide tartoznak például a H, V, B, D, G, M, É, Z, Á, O, I, S, R betűk.

Nagyon gyakori betűk: 5 pontot érnek. Ide tartoznak például az N, K, L, A, T, E betűk.

Inicializálás

A inicializal metódus létrehozza a letterPoints térképet, amely tartalmazza a közepesen ritka és a gyakori betűket a hozzájuk tartozó pontértékekkel.

A metódus a legritkabb betűk számára alapértelmezett pontértéket állít be, ha ezek a betűk nem találhatók a térképen.

Példa Pontszámításra

Példa 1: A játékos helyesen találja ki a K betűt, amikor még 8 próbálkozása van hátra.

A K betű egy nagyon gyakori betű, ezért 5 pontot ér.

Az összpontszám 5 ponttal nő.

Példa 2: A játékos helytelenül találja ki a Z betűt, amikor még 6 próbálkozása van hátra.

A Z betű közepesen ritka, de mivel a válasz helytelen, a próbálkozások száma alapján -2 pontot von le.

Az összpontszám 2 ponttal csökken.

Példa 3: A játékos helyesen találja ki a Q betűt, amely nem szerepel a letterPoints térképen.

A Q betű legritkább betűnek számít, ezért 15 pontot ér.

Az összpontszám 15 ponttal nő.

Következtetés

A pontrendszer a helyes és helytelen válaszokat dinamikusan kezeli, figyelembe véve a betűk gyakoriságát és a hátralévő próbálkozások számát. Ez a rendszer biztosítja, hogy a játékosok teljesítménye igazságosan legyen értékelve, és ösztönzi őket a ritkább betűk kitalálására, miközben bünteti a helytelen próbálkozásokat a játék előrehaladtával. A pontrendszer így jelentős szerepet játszik a játékélmény fokozásában és a versenyszellem fenntartásában.