

TESZTELÉS ÉS JAVÍTÁS DOKUMENTÁCIÓ

Tartalomjegyzék

1 Előszó	2
2 Hibajavítás #1	2
3 Hibajavítás #2	2
4 Hibajavítás #3	2
5 Hibajavítás #4	3
6 Hibajavítás #5	3
7 Hibajavítás #6	3
8 GitHub Használata	3

1 Előszó

A projektben, Rezák Kevin dokumentumszerkesztőként, valamint Bálint Vince hibakeresőként vett részt. A módosítások nyomon követését a GitHub platform biztosította.

2 Hibajavítás #1

Az első probléma a fájl betöltését végző metódusban (`LoadFromFile`) jelentkezett. Egy feltételes elágazás, amelynek feladata a program összeomlásának megelőzése lett volna, valójában folyamatos hibát okozott szemantikai hiba¹ miatt. A vizsgálat a szélesség (`Width`) tulajdonságot nézte a magasság (`Height`) helyett, emiatt hibát dobott a kód.

Megoldási javaslat (C#)

```
// Régi: if (lines.Length != Width) // Új: if (lines.Length != Height)
```

3 Hibajavítás #2

A második hiba szintén a betöltés során bukkant fel: a sorok feldarabolásakor a '|' karakter helyett szóköz alapján kellett volna tagolni, mivel a bemeneti fájlban szóközzel vannak elválasztva az RGB értékek.

Megoldási javaslat (C#)

```
// Régi: string[] values = lines[y].Split(';'); // Új: string[] values = lines[y].Split(' ');
```

4 Hibajavítás #3

A harmadik hiba a színek beállításánál jelentkezett a betöltéskor. Az index változó nem növekedett, így minden szín ugyanazt az értéket kapta a sorból. A probléma az index növelésének hiánya volt.

Megoldási javaslat (C#)

```
// Régi: byte.Parse(values[index]); // Új: byte.Parse(values[index++]);
```

¹Olyan hiba, amelyet emberi elírás okoz, de a fordító nem jelzi.

5 Hibajavítás #4

A negyedik probléma a MainWindow.xaml.cs fájlban fordult elő a pixel színének beállításakor. Szintaktikai² hiba történt a nem megfelelő színkonstruktur hívása miatt, illetve szükség volt a byte típusú konverzióra.

Megoldási javaslat (C#)

```
// Régi: Color.FromArgb(p.R, p.G, p.B) // Új: Color.FromArgb((byte)p.R,
(byte)p.G, (byte)p.B)
```

6 Hibajavítás #5

Aötödik hiba szintén a MainWindow.xaml.cs-ben volt: a kétdimenziós tömb indexei felcserélődtek a pixel kiolvasásakor, emiatt rossz helyről olvasta be az értéket.

Megoldási javaslat (C#)

```
// Régi: Pixel p = image.Pixels[y, x]; // Új: Pixel p =
image.Pixels[x, y];
```

7 Hibajavítás #6

A hatodik hiba szemantikai jellegű volt, és a pixelek elhelyezésénél jelentkezett. Az x változó a sorokon, az y az oszlopokon futott végig, de a Canvas.SetLeft hibásan x-et, a SetTop pedig y-t kapott.

Megoldási javaslat (C#)

```
Canvas.SetLeft(rect, x * CellSize); Canvas.SetTop(rect, y * CellSize);
```

8 GitHub Használata

A projekt során intenzíven használtuk a GitHub funkcióit. Kulcsfontosságú volt a branch-ek alkalmazása: minden fejlesztő a saját ágán dolgozott, így egy súlyos hiba sem ronthatta el a fő ágat. A kész változtatásokat Pull Request formájában küldték be, ahol a csapattagok átnézték a kódot. Hibamentes és elfogadott esetben Merge segítségével egyesítették a módosításokat a fő branch-be.

²Olyan hiba, amelyet a fordító is jelez.