

Név: Füzesiné Dr Hudák Mariann

Neptun kód: I97122

A programozás alapjai 1. nagy házi feladat dokumentáció (félkész)

Feladat: Képfeldolgozás

A program célja röviden (részletezést ld. a specifikációban):

A program *.ppm formátumú fájlt tölt be és végez rajta a felhasználó által választott módosításokat, majd a felhasználó választása szerint elmenti, vagy felülírja az eredeti képet.

A program jelenlegi állapota, funkciói:

A program jelenleg 3 modult tartalmaz:

- main
- readimage
- writeimage

Mindegyik modulhoz azonos nevű *.c és *.h fájl is tartozik.

A fentiekén kívül a programnak része az Infoc portálról letöltött debugmalloc.h fájl is, amely a dinamikus memóriakezelés helyességét ellenőrzi.

Main modul

A main modul felel a főmenü és az almenü megjelenítéséért és a program vezérléséért.

A main.h tartalmazza a főmenü és az almenü típus definícióit. A main.c displaymain() függvénye megjeleníti a főmenüt.

A selectmain() függvény bekéri a felhasználótól az alkalmazni kívánt képmódosításnak megfelelő menüpontot, és a választott menüpontot visszaadja a hívónak. A program jelen állapotában még mindegy, hogy a felhasználó melyiket választja, módosítást még nem végez a program a képen.

Ha a felhasználó nem létező menüpontot választ, a program figyelmezteti a hibára.

A main() függvény hívja a fenti két függvényt, majd a forráskódba beleírt auto.ppm fájl nevének paraméterként történő átadásával hívja a readimage modulban található readfile() függvényt a fájl beolvasásához, ennek sikere esetén a képernyőn visszajelez a fájl sikeres beolvasásáról, illetve hiba esetén a hibáról.

A hívott readimage fájl visszaadja a main-nek a fájlból kiolvasott short int-ekből álló pixelmátrixot.

A `writeimage()` függvény hívásával menti a kiolvasott fejléceket és pixelmátrixot bináris fájlként a forráskódba írt `PROBA.ppm` fájlnevével.

A program végleges verziójában a felhasználó tetszés szerint adhat majd meg egy `*.ppm` fájlt, amit módosítani szeretne, és szintén tetszőleges néven mentheti majd azt.

Readimage modul

A `readimage.h` 2 típust deklarál: a `PixelData` típus shortként tartalmazza a fájlból belvasott, egyes pixelekhez tartozó R, G és B adatokat.

Az `ImageData` típus a fájl fejlécéből beolvasott felbontás és bitmélység adatokat tartalmazza. A program 255 bitmélységű képfájlokat tud kezelni jelenleg, és a végleges program is 255 bitmélységet fog csak tudni kezelni.

A `readimage.c` első függvénye, a `readfilename()` bekér egy fájlnevet a felhasználótól, annak a `*.ppm` képnek a fájlnevét, amit a felhasználó módosítani szeretne. Ez a függvény jelenleg még nem kerül meghívásra.

A `readfile()` függvény olvassa be a kiválasztott (egyelőre forráskódba fixen beírt) `*.ppm` bináris fájlból az adatokat. A függvény a bináris fájlból kiolvasott pixelmátrix pointerével tér vissza, mely dinamikusan foglalt memóriaterületen található 2 dimenziós `PixelData` mátrixra mutat. Ez tartalmazza a kép minden egyes pixelének R, G és B értékét 0..255 skálán. A beolvasáskor történő esetleges hiba kódját a függvény a paraméterei között található short típusra mutató pointer által jelölt memóriacímre írja, ahonnan a hívó ki tudja olvasni az esetlegesen felmerülő hibakódot. Ha a függvény hiba nélkül lefut, az `errorcode`-ba 0-át ír.

A függvény karakterenként olvassa be a fájlt. Beolvasáskor először megnézi, érvényes `*.ppm` fájl kezdett-e beolvasni: ehhez a fájl fejlécében az első 2 karakternek `P6`-nak kell lennie. Ha nem ezt találja ott, hibát jelez és visszatér. Siker esetén szintén visszajelzést ad a szabványos kimeneten.

Ezután kiszűri az esetleges kommentet, amely `#` karakterrel kezdődhet, ha van, bármilyen hosszú lehet, és bármilyen karaktert tartalmazhat. Ezért az esetlegesen jelenlévő komment végjeléig (`'\n'`) olvas a kommentet kiszűrő kódrész.

Ezután beolvassa a kép felbontását. Ha nem volt komment, akkor a `size` első karakterébe `\n` kerül, de ezt a számmá való átkonvertáláskor a `sscanf` függvény figyelmen kívül hagyja, tehát nem befolyásolja a beolvasás sikerét.

Ha karakterként beolvasta a felbontást, a `sscanf` függvénnyel kinyeri a felbontás adatait számmá konvertálva azt. Ebből már tudja a függvény, mekkora memóriaterület foglalására lesz szükség a pixeladatok eltárolásához. A bitmélységet is hasonló módon olvassa be.

Ezután a kép felbontásának megfelelő méretű memória dinamikus foglalása történik meg a `heap`en a `malloc` függvény kétszeri hívásával: az első tömb a teljes adatsort tartalmazza (`matrix[0]`), a második pedig a 2 dimenziós tömb első oszlopában található tagok címeit; ezáltal áll elő a 2 dimenziós tömb.

A pixeladatokat 3 egymásba ágyazott ciklussal olvassa be. A belső ciklus az r, g és b értékeket nyeri ki az egyes pixelekről, a középső ciklus az egyes sorok minden elemén végigmegy, a külső ciklus pedig az egymást követő sorokon. Így a dinamikusan foglalt memóriaterületre beíródnak a kép pixeleinek adatai egy 2 dimenziós PixelData mátrixba. Ennek a mátrixnak a pointerét adja vissza a függvény, amin ezután a program a későbbiekben tetszőleges mátrixműveletet végezhet.

Writeimage modul

Ez a modul fájlba írja ki előbb a (későbbi programban már a feldolgozott) képet. Először előállítja a *.ppm fájl fejlécét. Ezután az olvasáshoz hasonló, 3 egymásba ágyazott ciklussal kiírja bináris fájlként az adatokat a *.ppm fájlba (jelenleg PROBA.ppm, a forráskódba írt fájlnevével).

A fájlba írás után a main() függvény felszabadítja a 2D pixelmátrix tárolására használt memóriaterületet.