**Dokumentacja PRI projekt 2 (Bartłomiej Binda)**

**Polecenie:**

Napisać program zarządzający zbiorem obrazów. Główne funkcje program: **dodawanie obrazu, kasowanie obrazu, zmiany dowolnego parametru, wyświetlanie listy obrazów, sortowanie obrazów przy zmiennym kryterium.** Obraz charakteryzujemy przy pomocy: wysokości obrazu, szerokości obrazu [w pikselach], typu {B8, B16, B32, C24, C16}, unikalnej nazwy (!!) oraz cech charakterystycznych obrazu: średnia znormalizowana intensywność (0-255), kolor dominujący (wektor [L,a,b]), histogram znormalizowany (wektor 8 elementowy). Przyjąć maksymalną liczbę obrazów równą 100.

**Wykonanie:**

Program wyświetla menu, gdzie do wyboru użytkownik ma wszystkie główne funkcje programu zadane w poleceniu. Menu wykonane jest za pomocą instrukcji switch case i zmiennej pomocniczej (option), która podawana jest przez użytkownika. Po wprowadzeniu zmiennej program odwołuje się do różnych przypadków, w każdym z nich wywołując jedną z danych funkcji, które opiszę w dalszej części tej dokumentacji.

W całym projekcie używam tablicy struktur do przechowania danych o każdym obrazie. Dodatkowo używana jest tablica o tej samej wielkości (#define SIZE), która dla każdej komórki z tablicy struktur analogicznie przechowuję 1, jeżeli jest w niej obraz, 0, jeżeli obraz tam nie istnieje, więc domyślnie tab składa się z samych zer. (int tab[SIZE] = {0}).

**Główne funkcje programu:**

* **Dodawanie obrazu:**

Dodawanie obrazu zrobione jest za pomocą dwóch funkcji pomocniczych.

**int sprDost(int tab[], int size):** funkcja daje numer pierwszej wolnej komórki do przechowania obrazu. Jeżeli wszystkie są pełne to f. (będę się tak odnosił do funkcji) oddaje -1.

**void dodObraz(obraz \*ob):** funkcja pobiera wszystkie parametry obrazu od użytkownika i zapisuje je w danej komórce.

* **Kasowanie obrazu:**

**void kasObraz(int \*tab, obraz \*ob, int komorka):** funkcja kasuje wybrany obraz z tablicy obrazów. Jeżeli nie istnieje obraz w tej komórce, poinformuje o tym użytkownika.

* **Zmiana dowolnego parametru:**

**void zmienParametr(obraz \*ob):** f. zmienia dowolny wybrany parametr wybranego obrazu. Jeżeli obraz nie istnieje informuje o tym użytkownika. (wybór parametru za pomocą instrukcji switch case).

* **Wyświetlanie obrazu:**

**void czytObraz(obraz ob):** Jeżeli obraz istnieje, wyświetla wszystkie informacje o danym obrazie.

* **Sortowanie obrazów (wszystkie malejące):**

**void sortWys(obraz \*tablica, int tab[], int size):** sortuje wszystkie obrazy względem wysokości

**void sortSzer(obraz \*tablica, int tab[], int size):** sortuje wszystkie obrazy względem szerokości

**void sortWek(obraz \*tablica, int tab[], int size):** sortuje wszystkie obrazy względem wektora koloru dominującego

Wszystkie funkcje sortujące pod koniec posługują się funkcją pomocniczą:

**void zmienTab(obraz \*tablica, int tab[], int size):** funkcja modyfikuje tablice tab z 1 i 0, aby się zgadzała z istniejącymi obrazami.

**Przykład:** obrazy istnieją w komórkach 1, 2 i 4. Po sortowaniu będą posortowane w komórkach 1,2 i 3 więc tablica dostępności musi zostać zmodyfikowana

* **Funkcja pomocnicza czysc():** czyści bufor

**Wykonanie: Bartłomiej Binda**

**Termin oddania: 14.11.2018**