

# KEYPIANO

Filip Gawęł

## 1. Część I

### • Opis programu

keypiano jest prostym (pracującym w konsoli) programem imitującym keyboard. Na ekranie widoczne jest menu oraz klawiatura pianina wraz z odpowiadającymi do niej klawiszami klawiatury 'qwerty'. Program wydobywa odpowiednie dźwięki trwające 100 milisekund. Zakres częstotliwości dźwięków: od  $C^1 = 261 \text{ Hz}$  do  $FIS^2 = 740 \text{ Hz}$ . Fortekeypiano nie ma możliwości odtworzenia więcej niż jeden dźwięk w tym samym czasie. Do obsługi programu wystarczy komputer z systemem operacyjnym Windows XP lub nowszym.

### • Instrukcja obsługi

Do poprawnego funkcjonowania potrzebujesz pliku "piano.exe", a także dwóch plików tekstowych: "zapis.txt" oraz "sto\_lat.txt". Wszystkie pliki powinny znajdować się w tym samym miejscu.

Aby zagrać na keypiano uruchom program "piano.exe". Do poruszania się w menu używaj liczb z zakresu od 1 do 5, strzałka sygnalizuje Ci z jakiego trybu aktualnie korzystasz.

#### WYGLĄD GŁÓWNEGO MENU

```
MENU:
1. Tryb wolny<--
2. Nagrywaj
3. Posłuchaj
4. Odtwórz ostatnie nagranie
5. Wyjdź z programu

| w | e |   | t | y | u |   | o | p |   |
|_ |_ |_ |_ |_ |_ |_ |_ |_ |_ |_ |_
| a | s | d | f | g | h | j | k | l | ; | ' | \ |
|_ |_ |_ |_ |_ |_ |_ |_ |_ |_ |_ |_

Graj...
```

#### Tryb wolny:

W celu wydobywania konkretnego dźwięku naciśnij odpowiadający mu klawisz z klawiatury komputerowej.

#### Nagrywaj:

Od tego momentu nagrywasz swój utwór wciskając odpowiednie klawisze. Gdy zakończysz granie utworu naciśnij klawisz 2 – STOP.

#### Odtwórz ostatnie nagranie:

W tym momencie program odtworzy nagrany przez Ciebie utwór.

**Posłuchaj:**

Z głośników wydobywa się znana muzyka nagrana przez programistę. Aby grać dalej zaczekaj, aż utwór zakończy się.

**Wyjdź z programu:**

Zostaniesz zapytany czy na pewno chcesz zakończyć działanie programu. Gdy nie chcesz opuścić program naciśnij 'n', gdy jednak chcesz wyłączyć program naciśnij 't'.

**Lista komunikatów oraz ich znaczenie:**

KOMUNIKAT	ZNACZENIE
„Graj...”	Gdy naciśniesz na odpowiedni klawisz na klawiaturze 'qwerty' program odtworzy dźwięk.
„Nagrywam”	Każdy naciśnięty przycisk zostanie nagrany
„Nic nie nagrales”	Nie nagrałeś jeszcze utworu.
„Zapisywanie utworu...”	Zaczekaj, aż program zapisze twój utwór
„STOP zapisywanie...”	Podczas nagrywania utworu chciałeś opuścić tryb nagrywania, musisz poczekać aż program zapisze nagranie
„Słuchaj...”	Jesteś proszony o słuchanie muzyki, w tej chwili nic nie możesz naciśnąć
„Brak pliku: sto_lat.txt”	Błąd polegający na tym, że w folderze z aplikacją nie ma pliku o podanej nazwie. Poproś autora programu o pomoc
"Brak pliku zapis.txt"	Błąd polegający na tym, że w folderze z aplikacją nie ma pliku o podanej nazwie.
„Odtwarzanie nagrania”	W tym momencie możesz posłuchać swoje ostatnie nagranie
"Jestes pewien, ze chcesz zakonczyc prace z programem? T/N"	Być może nieumyślnie naciśnąłeś przycisk wyjścia, jeśli nie chcesz opuszczać programu naciśnij 'n'(nie). Natomiast, gdy chcesz opuścić program naciśnij 't'(tak).
"Bledny wybor"	Jesteś proszony o naciśnięcie klawisza 't' – wyjście z programu lub 'n' – nie wyłączaj programu.

## 2. Część II

### • Specyfikacja techniczna

(a) Podział projektu na pliki:



(b) Co znajduje się w danym pliku

pianino.c – główny program

(c) Używane biblioteki

- `stdio.h`
- `windows.h`
- `time.h`

(d) Kompilacja projektu Do stworzeniu projektu został wykorzystany CodeBlocks w wersji 16.01 oraz kompilator GNU GCC.

### • Szczegóły techniczne

a) Działanie programu

Na początku program przechodzi do funkcji *rozpoczęcie()* jest ona bezargumentowa typu void. Program wyświetla menu oraz poprzez funkcję *pianino()* wyświetla klawiaturę pianina. Pod klawiaturą użytkownik jest na bieżąco informowany czego oczekuje program.

Użytkownik po naciśnięciu w klawiaturę komputerową natychmiast dzięki funkcji *getch()* przypisuje wartość do zmiennej globalnej *klawisz* typu char. Z głośników wydobywa się dźwięk za pomocą funkcji *dzwiek()*. W danej funkcji program w zależności od wartości zmiennej *klawisz* odtwarza dźwięk o specjalnie dobranej częstotliwości, który trwa 100 milisekund. Wszystko za sprawą funkcji Beep dostępnej w bibliotece windows.h. Gdy do zmiennej *klawisz* zostanie przypisana cyfra z przedziału od 1 do 5 program przejdzie do konkretnej funkcji wskazanej przez funkcję *switch()*.

Gdy użytkownik wybierze cyfrę '2' z klawiatury komputerowej program przejdzie do funkcji *nagrywaj()*. Ta funkcja zapisuje do pliku *zapis.txt* naciskane klawisze *fprintf(fzapis, "%c\n", klawisz)* oraz czas pomiędzy naciśnięciami. Czas liczony jest od momentu, kiedy już pierwszy klawisz został naciśnięty dlatego zmienna *i = -1*, gdyż dzięki warunkowi *i >= 0* program nie liczy czasu za pierwszym razem. Do pliku w kolejnych liniach odpowiednio wpierw zapisywana jest wartość *klawisz* potem: *stop - start - czasTrwaniaDzwieku*. Gdzie *czasTrwaniaDzwieku = 100*.

Po wybraniu przez użytkownika cyfry '3' program przejdzie do funkcji *posluchaj()*. Dana funkcja odczytuje plik *sto\_lat.txt* najpierw przypisując pierwszą linijkę do *klawisz*, a drugą do *czasOdstepu* i za pomocą funkcji *Sleep(czasOdstepu)* program zasypia na odpowiedni czas tworząc tym samym pauzę.

Gdy użytkownik naciśnie klawisz z numerem '4' program przejdzie do *odczyt()*. Dana funkcja odczytuje plik *zapis.txt* i w taki sam sposób jak w funkcji *posluchaj()* odtwarza dźwięk.

Wczytanie przez program liczby '5' spowoduje przejście do funkcji *koniec()*, w której zostanie użytkownikowi zadane pytanie, czy na pewno chce opuścić program.

## b) Algorytmy/fragmenty kodu

### Algorytm odpowiadający za sprawdzenie czy plik tekstowy jest pusty

```
fseek(fzapis, 0, SEEK_END);
unsigned long len = (unsigned long)ftell(fzapis);
printf("\nOdtwarzanie nagrania\n");
if (len > 0) //sprawdza czy plik jest pusty
{
    rewind(fzapis);
    while(!feof(fzapis))
    {
        fscanf(fzapis, "%c\n", &klawisz); //odczytuje pliku
        dzwiek();
        fscanf(fzapis, "%d\n", &czasOdstepu);
        Sleep(czasOdstepu);
    }
}
else
{
    printf("Nic nie nagrales!\n");
    Sleep(1000);
}
rozpoczecie();
```

<i>SEEK_END</i> :	przesuwa położenie wskaźnika pliku na koniec pliku.
<i>ftell()</i> :	funkcja zwraca aktualną pozycję wskaźnika pliku.
<i>fseek ()</i> :	służy do przesuwania pozycji wskaźnika pliku do podanej lokalizacji.
<i>rewind()</i> :	funkcja przewija plik do początku