

1. Instrukcja

Po uruchomieniu gry, zobaczymy 3 możliwości

- play – Rozpoczyna grę.
- scoreboard – Wyświetla tabelę ośmiu najlepszych wyników.
- exit – Przerywa program.

Nawigacja po menu odbywa się za pomocą strzałek (góra, dół), wybór zatwierdzany jest klawiszem ENTER.

Gra polega na trafianiu w chomiki wyskakujące z norek. Strzał odbywa się przez naciśnięcie przycisków X, C, V, B, N, M odpowiadających odpowiednim norkom licząc od lewej strony. W momencie trafienia, graczowi przyznawany jest punkt. Gdy nie trafimy, lub chomik zdąży się schować zanim zareagujemy, odejmowane jest jedno życie (początkowo gracz zaczyna z 20).

W momencie przebicia jednego z ośmiu najlepszych wyników, gracz może wpisać się na listę.

2. Opis poszczególnych plików

main.cpp – Główny plik z kodem.

Hamster.h – Plik nagłówkowy struktury reprezentującej chomika.

- `void _hit(); void _up(); void _hide(); void _miss()` – manipulatory stanu
- `int get_status()` – zwraca aktualny stan chomika
- `void print_line(int)` – drukując wybraną linię „grafiki” chomika
- `void tidy()` – ukrywa krety które nie mają być już dłużej pokazywane
- `void peek_a_boo()` – realizuje losowe wyskakiwanie chomika z norki
- `bool check_for_timeout()` – sprawdza czy chomik nie jest pokazywany już za długo, jeśli tak to ukrywa go
- `bool punch()` – sprawdza czy chomik został trafiony (czy jest na zewnątrz w momencie uderzenia)

ScoreTable.h – Plik nagłówkowy struktury reprezentującej tabelę wyników.

- `Record` – struktura pomocnicza, przechowująca pojedynczy wpis tabeli wyników
- `void swap(Record*, Record*)` – funkcja pomocnicza dla sortowania, zamienia dwa wpisy miejscami
- `void bubble_sort(Record [], int)` – funkcja pomocnicza w której zaimplementowany jest algorytm sortowania bąbelkowego
- `void read()` – wczytuje dane z pliku i ładuje je do struktury tabeli wyników
- `void sort()` – sortuje tabelę wyników malejąco według punktów
- `int lowest_score()` – zwraca wartość najniższego wyniku w tabeli
- `void push_record(string name, int score)` – dodaje wpis do tabeli
- `void print()` – wyświetla tabelę

content.h – Plik z nagłówkami funkcji pomocniczych

- `int` get_key_index(`char`) – sprawdza jakiemu numerowi chomika odpowiada wciśnięty przycisk
- `char` get_key() – sprawdza czy bufor klawiatury nie jest pusty, jeśli nie, to zwraca pobrany znak
- `void` print_base() – drukuje norki
- `int` print_menu(`int`) – drukuje menu z zaznaczonym wyborem który pobiera jako argument

external.h – Plik z nagłówkami funkcji zewnętrznych, tj nie napisanych przezemnie, odnoszących się do systemu.

- `void` setFontSize(`int`) – ustawia rozmiar czcionki
- `bool` setWindowSize(`size_t`, `size_t`) – ustawia wymiary okna
- `void` goToXY(`int`, `int`) – ustawia kursor na wybranej pozycji
- `void` setColor(`int`) – ustawia kolor tekstu w konsoli
- `void` hideCursor() – ukrywa kursor

Wszystkie powyższe pliki nagłówkowe posiadają swoje odpowiedniki .cpp, poniżej pliki zawierające stałe

GRAPHICS.h – Plik zawierający stałe „grafiki” złożone ze znaków

CONSTANTS.h – Plik zawierający stałe wykorzystywane w kodzie

Scoretable.txt – Plik przechowujący tabelę wyników