

Temat: Platformowa gra 2D

Autor: Jakub Grobelny

1. Główne cechy:

- Platformowa gra 2D polegająca na poruszaniu się w prawo do końca poziomu, unikaniu/pokonywaniu przeciwników i skakaniu po platformach.
- Grafika 2D wyświetlana przy pomocy biblioteki Allegro (wersja 5.2.3).
- Fizyka 2D:
 - Grawitacja.
 - Tarcie pomiędzy postacią i powierzchnią (pozwala to na dodanie podłoża, po którym gracz się ślizga).
- Destrukcja niektórych elementów otoczenia (niszczenie klocków poprzez uderzanie w nie głową tak jak w Mario).
- Przeciwnicy wzorowani będą na tych z gry Mario i dzielić się będą na trzy typy:
 - Przeciwnicy poruszający się po ziemi:
 - "Żółwie" - poruszają się od przeszkody do przeszkody. Po zabiciu zostaje po nich muszla, która po popchnięciu ślizga się po podłożu i może zabić gracza.
 - Zwykli przeciwnicy, którzy poruszają się tak jak żółwie ale wpadają w przepaście. Wystarczy raz na nich wskoczyć aby ich wyeliminować.
 - Przeciwnicy latający:
 - Latające "żółwie". Poruszają się w jednym kierunku po torze wyglądającym jak wykres funkcji $|\sin(x)|$. Po strąceniu z powietrza tracą skrzydła i zmieniają się w zwykłe żółwie.
 - Przeciwnicy statyczni:
 - Mięsożerne kwiaty. Zjadają gracza po kontakcie od góry.
 - Armaty wystrzeliwujące pociski.
- Do gry dołączony będzie edytor poziomów pozwalający na ustawianie poszczególnych elementów na planszy i zapisywanie jej do formatu pliku obsługiwanego przez grę.
- Gracz będzie mógł zbierać przedmioty dające mu bonusy takie jak chwilowa nieśmiertelność.

2. Organizacja kodu:

- W programie w dużym stopniu będą wykorzystywane struktury. Na każdą z nich przeznaczone będą oddzielne pliki .h oraz .c. Planowane są następujące główne struktury:
 - **Object** - struktura przechowująca pozycję, typ, grafikę (razem z animacją), hitboxy oraz właściwości fizyczne danego obiektu. Przeznaczona będzie do reprezentowania dowolnego typu obiektu, od platform do postaci gracza.
 - **Hitbox** - struktura reprezentująca figurę (prostokąt albo koło), która służy do wykrywania kolizji pomiędzy obiektami.
 - **Physics** - zbiór właściwości fizycznych danego obiektu. Struktura ta przechowuje wektory prędkości obiektu, współczynnik tarcia, masę itd.
 - **ObjectsList** - struktura będąca dynamiczną listą struktur typu **Object**. Używana będzie między innymi do sprawdzania, czy zachodzi kolizja z którymkolwiek z obiektów znajdujących się na mapie.
- Plik **defines.h** zawierający definicje kolorów w RGB, rozdzielczości gry, ilości klatek na sekundę itp.
- Pliki **keyboard.h** i **keyboard.c** odpowiedzialne za obsługę klawiatury (aktualizowanie stanu klawiszy).
- Pliki **player.h** i **player.c**, w których znajdują się funkcje związane z postacią gracza (ruch, skakanie itp.).
- Pliki **util.c** i **util.h** zawierające pomocnicze, uniwersalne funkcje, które mogą być przydatne w różnych obliczeniach.
- Plik **main.c**, w którym znajduje się główna pętla gry.

3. Interakcja użytkownika z programem:

- Gracz będzie sterował postacią za pomocą przycisków **góra, dół, lewo, prawo** lub **W, S, A, D**, które będą kolejno odpowiedzialne za skok, kucnięcie, ruch w lewo i ruch w prawo. Skakać będzie można również za pomocą **spacji** a kucać za pomocą przycisku **ctrl**.
- Menu główne oraz menu pauzy będzie można obsługiwać zarówno za pomocą myszy jak i klawiatury.
- Rozmieszczanie obiektów w edytorze realizowane będzie przy użyciu myszy.