Programowanie zdarzeniowe

Dokumentacja do projektu

Krzysztof Najda

Politechnika Warszawska

Wydział Elektroniki i Technik Informacyjnych

Zawartość projektu

Opracowany przeze mnie program imituje grę Atari – Breakout, w której to gracz ma za zadanie zbić wszystkie bloczki za pomocą kulki i sterowalnej paletki.

Koncepcja świata gry

Tabela interakcji

	Kulka	Bloczek	Paletka	Bonus	Granice
Kulka	-	X	X		X
Bloczek	X	-			
Paletka	X		-	X	
Bonus			X	-	X
Granice	X			X	-

X – oznacza że pomiędzy elementami zachodzą jakieś interakcje

Kulka

Przywoływana przez gracza lub za pomocą aktywacji bonusu.

Po wykryciu zderzenia z bloczkiem, granicami świata lub paletką, jest odbijana symetrycznie względem wektora normalnego powierzchni z którą zaszło zderzenie. Wyjątkiem od tej reguły jest odbicie od górnej krawędzi paletki (kąt odbicia zależy od odległości kulki od środka paletki), lub dolnej krawędzi ekranu (kulka jest usuwana).

Bloczek

Powoływany przy załadowaniu świata gry. Może być niezniszczalny lub mieć pewną wytrzymałość. Jeśli bloczek nie jest niezniszczalny to po zderzeniu z kulką jego wytrzymałość jest dekrementowana. Gdy wytrzymałość osiągnie zero, bloczek jest niszczony.

Po zniszczeniu bloczka istnieje pewna szansa na powołanie bonusu.

Paletka

Powoływany przy załadowaniu świata gry. Jest to jedyny element świata gry na który gracz ma bezpośredni wpływ – może go przesuwać w lewo lub prawo.

Jeśli nastąpi zderzenie kulki z górną krawędzią paletki to kąt odbicia kulki zależy od jej odległości od środka paletki – na środku paletki jest to kąt prosty, a przy jej krawędziach bocznych kąt jest najostrzejszy.

Jeśli nastąpi kolizja bonusu z paletką to bonus ten jest aktywowany i usuwany. Granice świata nie ograniczają paletki – tj. może ona wyjechać częściowo lub całkowicie poza ekran.

Bonus

Powoływany po zbiciu bloczka.

Po powołaniu bonusu, zaczyna on spadać w stronę dolnej krawędzi ekranu. Jeśli nastąpi zderzenie z tą krawędzią to bonus jest usuwany bez aktywacji. Jeśli jednak zdąży przechwycić go paletka, to jest on aktywowany.

Przykładowe efekty aktywacji bonusu to: powołanie kolejnych kulek, przyspieszenie paletki lub zwiększenie promienia kulek.

Funkcjonalności

Bonusy

Zaimplementowane zostały następujące bonusy:

- Poszerzenie paletki
- Zwiększenie promienia kulek
- Powołanie na raz 3 standardowej wielkości kulek
- Powołanie na raz wielu małych kulek
- Wystrzeliwanie z krawędzi paletki wielu małych kulek w równych odstępach czasu, usuwane przy pierwszej kolizji z bloczkiem lub granicami okna
- Przyspieszanie kulki
- Przyspieszanie paletki

Wszystkie te bonusy są aktywowane czasowo, a więc znikają po pewnym czasie

Wczytywanie plansz

Plansze do rozgrywki można wczytywać z uprzednio przygotowanych plików. Naturalnie można je również zapisywać do pliku jednak nie jest to możliwe z poziomu gracza.

Tworzenie statystyk

Przy każdej rozgrywce tworzone są statystyki które zawierają następujące informacje:

- Nazwę użytkownika
- Uzyskany wynik
- Czas rozpoczęcia gry
- Czas trwania gry w sekundach

Przy czym do jednej statystyki mogą się wliczać wyniki z rozgrywki z wielu plansz o ile były one rozgrywane przez tego samego gracza w jednym ciągu. Inaczej mówiąc wyniki są liczone do tego samego rekordu od włączenia menu wyboru planszy do wyjścia do menu przez jednego użytkownika.

Zawartość pakietów

W trakcie implementacji zostały utworzone 4 pakiety klas: game_objects, breakout_proper, breakout_manager oraz breakout_interface. W tej części omówię każdy z nich.

Game_objects

Pakiet ten trzyma najbardziej podstawowe obiekty z pozostałych trzech innych pakietów i jest dla nich swego rodzaju bazą.

Point2d – reprezentuje punkt w przestrzeni 2d. Użyteczny w implementacji wszystkich obiektów geometrycznych.

Vector2d - reprezentuje wektor na przestrzeni 2d. Użyteczny przy poruszaniu obiektów geometrycznych oraz rozróżnianiu krawędzi prostokątów na których nastąpiła kolizja (wektor normalny)

Rectangle2d – reprezentuje prostokąt na przestrzeni 2d. Jest bazą dla takich obiektów jak bloczek, bonus czy paletka.

Circle2d – reprezentuje koło na przestrzeni 2d. Jest bazą dla kulki.

Collision2d – zbiór metod za pomocą których można rozpoznawać kolizje pomiędzy obiektami geometrycznymi oraz ustalać wektory normalne takich kolizji.

Breakout_proper

Zawiera właściwą część gry, czyli takie obiekty znane z części koncepcyjnej jak kulka czy blok, ale nie tylko.

Ball, Block, Bonus, Paddle – reprezentują właściwe sobie obiekty znane z części koncepcyjnej.

BlockSet – jest zbiorem bloków, reprezentującym daną plansze. Dzieli obszar w siatkę tak że każdy bloczek musi taki sam wymiarowo, oraz nie mogą być ustawione asymetrycznie względem siebie (a więc mogą się stykać albo na całej długości krawędzi, albo na jednym wierzchołku albo wcale)

BallSet, BonusSet – są fabrykami dla kulek i bonusów odpowiednio. Tu są powoływane i trzymane wszystkie kulki i bonusy w świecie gry.

BonusDatabase – przetrzymuje efekty aktywacji bonusów oraz udostępnia je gdy powoływany jest bonus.

World – jest reprezentacją całego świata gry. Trzyma wszystkie powyższe obiekty oraz zajmuje się poruszaniem ich i rozwiązywaniem interakcji między nimi (czyli to tutaj rozwiązywane jest np. ruch kulki i jej zderzenie z bloczkiem).

Breakout_manager

W tym pakiecie obsługiwane są sytuacje które nie są bezpośrednio częścią świata gry, ale są niezbędne by zapewnić pewne funkcjonalności.

RestrictedWorld – dziedziczy po klasie World z pakietu omawianego poprzednio. Oprócz funkcji klasy World, liczy ona także wynik gracza oraz liczbę pozostałych mu żyć (jeśli ich zabraknie, klasa nie pozwala na powołanie kulki – stąd 'Restricted').

FileManager – zajmuje się serializacją i deserializacją planszy (BlockSet) na których można potem prowadzić rozgrywkę.

RecordManager – zajmuje się serializacją i deserializacją wyników gracza oraz prowadzeniem statystyk (np. kiedy zaczął grać i jak długo grał).

WorldManager – tak jak World łączył obiekty z wcześniej omawianego pakietu, tak WorldManager łączy klasy z obecnego pakietu. Umożliwia on ustawianie i resetowanie planszy oraz włącza i wyłącza RecordManagera.

Breakout_interface

Ostatni z pakietów zajmuje się obsługą zdarzeń i wyświetlaniu gry na ekranie.

GameGraphics – zajmuje się rysowaniem obiektów, zleconych mu przez GameControler'a.

GameControler – obsługuje zdarzenia wykreowane przez gracza oraz daje dostęp do pozostałych funkcjonalności (takich jak podglądnie statystyk prowadzonych przez RecordManager'a).

Interfejs

Po uruchomieniu gry otworzy się okienko z menu.

W menu znajdują się 3 opcje – zacząć grę, zobaczyć wyniki oraz opuścić grę. Ponad tymi opcjami znajduje się miejsce na wpisanie swojej nazwy użytkownika pod którą zostaną zapisane wyniki.

Po kliknięciu przycisku 'PLAY', otworzy się okienko z listą możliwych do rozegrania plansz oraz dwa przyciski 'Select' – którym potwierdza się wybór mapy i przenosi do gry, oraz 'Back' którym można wrócić do menu.

W tym miejscu warto odnotować, że wyniki zaczynają być zapisywane od momentu kliknięcia przycisku 'PLAY' w menu, do momentu kliknięcia przycisku 'Back' w oknie wyboru planszy, przy czym nie będą one zapisane jeśli nigdy nie weszło się do właściwej gry (czyli kliknęło przycisku 'Select').

Podczas rozgrywki właściwej sterowanie odbywa się następującymi klawiszami:

A	Ruch paletki w lewo
D	Ruch paletki w prawo
R	Reset planszy
ENTER	Powołanie nowej kulki
ESC	Wyjście z rozgrywki do okna wyboru planszy