

Projekt „Labirynt”

Autor: Izabela Strumiecka

1. Analiza obiektowa.

Projekt jest tekstowa gra, która polega na znalezieniu wyjścia z labiryntu i nazbieraniu jak największej ilości złota. Chociaż jest to gra tekstowa, musi być uruchomiona w środowisku graficznym, z uwagi na sposób obsługi komunikacji z użytkownikiem przez klawiaturę. Tak obsługa jest konieczna ponieważ gra jest czasu rzeczywistego – dodatkowe postacie w grze są osobnymi wątkami.

Większość elementów gry tj. ilość złota, lekarstw, złodziei, bandytów, ułożenie korytarzy, nie jest z góry ustalona i jest tworzona podczas generowania labirynt – co daje niepowtarzalny i oryginalny charakter każdej gry.

Labirynt składa się z komnat, których w grze jest 2500. W każdej komnacie może znajdować się złoto (25, 50, 75 lub 100), lekarstwo (25, 50, 75 lub 100), broń (kamień, drąg lub sztylet), złodziej (zabierający bohaterowi część złota) oraz bandyta (zabierający bohaterowi część energii).

Opis klas:

Labirynt - główna klasa programu, Ma referencje do klasy Bohater oraz dwuwymiarową tablicę komnat. Zawiera metody między innymi do generowania labiryntu, wypisania mapy bohatera lub do zakończenia gry. Dziedziczy po klasie JFrame.

Klawisze - klasa odpowiedzialna za reagowanie na naciśnięcie klawisza na klawiaturze. Jeśli naciśnięty klawisz jest rozpoznany podjęta zostaje stosowna akcja. Dziedziczy po klasie KeyAdapter.

Bohater - klasa reprezentująca bohatera gry. Zawiera pola określające wartość zebranego złota, ilość energii, posiadaną broń, siłę bohatera oraz pozycję w labiryncie.

Komnata - reprezentuje pojedynczą komnatę w labiryncie. Posiada pola określające ilość złota w komnacie, ilość lekarstwa, rodzaj komnaty oraz pola mówiące czy w komnacie jest bohater, złodziej lub bandyta oraz czy dana komnata jest widoczna na mapie bohatera.

Bron - klasa zawierająca pola, które są dziedziczone przez klasy reprezentujące poszczególne bronie. Zawiera pola określające nazwę broni, jej siłę oraz trwałość.

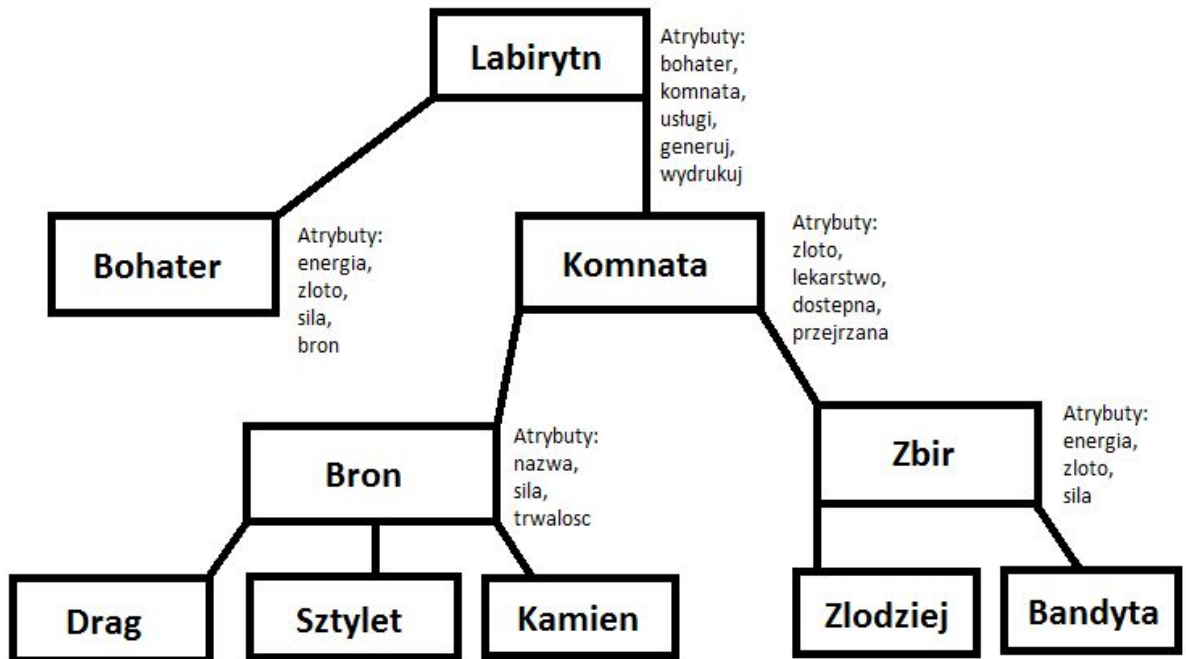
Kamien, Drag, Sztylet - klasy dziedziczące po klasie Bron, różniące się wartościami odziedziczonych pól.

Zbir - klasa zawierająca pola, które są dziedziczone przez klasy reprezentujące poszczególnych zbirów, dziedziczy po klasie Thread. Zawiera pola określające stan energii, ilość posiadanego złota oraz siłę zbira.

Złodziej - reprezentuje złodzieja, dziedziczy po klasie Zbir. Zawiera pola określające stan energii oraz ilość posiadanego złota. Złodziej kradnie bohaterowi złoto, jeśli ten jest w tej samej komnacie, podejmuje decyzję do jakiej komnaty się przenieść i czeka 2 sekundy.

Bandyta - reprezentuje bandytę, dziedziczy po klasie Zbir. Zawiera pola określające stan energii oraz siłę bandyty. Bandyta atakuje bohatera będącego w tej samej komnacie i zabiera mu energię. Może być przyczyną wcześniejszego zakończenia gry. Podejmuje decyzję o zmianie lokacji i czeka 2 sekundy.

2. Projekt obiektowy.



Przepływ danych kontrolowany jest przez klasę dziedziczącą po KeyAdapter. Klasa ta jest odpowiedzialna za reakcję na naciśnięcie klawisza na klawiaturze. Klasa ta poprzez referencje do klasy labirynt ma dostęp do innych klas i to ona odczytuje i zmienia wartości atrybutów poszczególnych klas.

3. Inne zastosowania.

Ze względu na charakter projektu zastosowanie jego klas jest ograniczone i zawęża się do gier. Klasy Bron oraz Zbir można użyć w dowolnym innym programie, w którym występuje obiekt broni. Mają one ogólną definicję i nie wymagają żadnych parametrów ani referencji przez co są uniwersalne.