

Politechnika Śląska
Wydział Matematyk Stosowanej
Kierunek Informatyka

Gliwice, 09.02.2022

Programowanie I

Projekt zaliczeniowy

" Szkarłat - tekstowa gra cRPG "

Kamil Ptak gr. lab. 8

1. Opis projektu.

Krótki zwięzły opis tematyki projektu

„Szkarłat” jest grą tekstową z gatunku cRPG, wzorowanej na grze „Otchłań. Gracz tworzy swoją postać, którą porusza się oraz wchodzi w interakcje ze światem i otoczeniem przy pomocy komend tekstowych. Z uwagi na konwencję w jakiej gra została stworzona, wszystkie opisy, rozmowy lub czynności są przekazywane w postaci tekstu, a nie za pomocą grafiki lub dźwięków. Gracz nie posiada odgórnego celu, bądź zadania, może do woli eksplorować oraz poznawać świat przedstawiony. Gracz jest w stanie rozwijać swoją postać, zdobywając kolejne poziomy doświadczenia oraz wydając zbierane złoto na lepsze wyposażenie. Jedynym sposobem w jaki gracz może przegrać grę jest utrata punktów życia do 0, w skutek przegranej w walce.

2. Wymagania

- Ekspansywność projektu – łatwość w dodawaniu nowych danych, przy minimalnej ingerencji bezpośrednio w kod gry;
- Ekran menu głównego z nazwą gry, umożliwiającego rozpoczęcie nowej gry, wczytanie poprzedniej rozgrywki i wyjście z gry;
- Proces tworzenia postaci gracza, umożliwiający nadanie dowolnej nazwy oraz wybór jednej z trzech odgórnie utworzonych klas;
- Interpreter komend użytkownika, pozwalający na bezpośredni wpływ na rozgrywkę;
- System ekwipunku, pozwalający na używanie, usuwanie oraz wyświetlanie dokładnego opisu każdego znajdującego się w nim przedmiotu oraz wyposażenia użytkownika
- Poruszanie się korzystając z oznaczeń kierunków świata oraz wyświetlanie opisów lokacji, umożliwiających zapamiętanie oraz rozrysowanie lokacji;
- Przeprowadzanie wymiany zdań z postaciami niezależnymi przy pomocy odgórnie utworzonych dialogów;
- System handlu, umożliwiający na wydanie zdobytego przez gracza złota i kupno nowych przedmiotów;
- Turowy system walki, w którym ruch podejmuje na zmianę gracz oraz jego przeciwnik (postać niezależna);

- System rozwoju postaci, który w miarę zdobywania przez gracza punktów doświadczenia zwiększa jego statystyki, oraz odblokowuje kolejne zdolności;
- Klasy postaci, które posiadają wyjątkowe dla siebie przedmioty oraz umiejętności;

3. Przebieg realizacji

3.1 Użyte biblioteki

- **String** – umożliwia na korzystanie z klasy string, służącej do operowania na ciągach znaków (tekst);
- **Windows.h** – biblioteka umożliwiająca na korzystanie z funkcje Sleep(), która została użyta w celu „opóźnienia” wypisywanego tekstu;
- **Locale.h i conio.h** – biblioteki użyte w celu wypisywania polskich znaków w konsoli
- **cctype** – zapewnia zbiór funkcji do klasyfikowania i transformowania indywidualnych znaków. W projekcie korzystałem z funkcji konwersji tolower i toupper.
- **vector** – biblioteka, którą najbardziej wykorzystywałem w projekcie. Umożliwia użycie wektora, czyli struktury danych reprezentujących tablicę. Na bazie wektorów działają wszystkie bazy zasobów z których korzysta mój projekt.
- **sstream** – zapewnia dostęp do klasy string stream, dzięki której zbudowany na potrzeby projektu interpreter rozdziela pobrane zdanie na pojedyncze słowa, które następnie osobną są odczytywane. Ponadto przy pomocy klasy stringstream dokonywałem konwersji z wartości intiger do string.
- **fstream** – biblioteka zapewnia szereg funkcji służących do zapisu i odczytu danych z i do pliku. Dzięki niej projekt pobiera podstawowe wymagane dane bezpośrednio z plików źródłowych i zapisuje postępy oraz statystyki każdego gracza.
- **random i ctime** – obie te biblioteki zostały wykorzystane w celu wygenerowania liczb pseudolosowych.

3.2 Pliki składowe projektu

“Szkarałat” składa się z dwóch głównych folderów:

- **saves**, który służy do przechowywania wszystkich zapisów użytkownika. Przy zaczęciu nowej gry tworzony jest plik tekstowy o nazwie takiej sam jak nazwa gracza, oraz (w przypadku kiedy nie istnieje) tworzony jest plik **saves-list.txt**, do którego dopisywana jest nazwa gracza. Plik **saves-list.txt** jest rejestrem wszystkich zapisów, wypisywanym przy wyborze pliku do wczytania.
- **sources**, który zawiera dwa pliki tekstowe:
 - **intro.txt**, zawierający nazwę gry zapisaną symbolami ‘\$’, wypisywaną podczas włączenia programu;
 - **tworzeniePostaci.txt**, zawierający tekst wyświetlany podczas tworzenia nowej postaci gracza.

Folder ten zawiera dwa kolejne podfoldery:

- **databases**, zawierający pliki **BronieBZ.txt**, **KlasyBZ.txt**, **LokacjeBZ.txt**, **NpcsBZ.txt**, **PrzeciwnicyBZ.txt**, **PrzedmiotyBZ.txt**, **UmiejetnosciBZ.txt** oraz **ZbrojeBZ.txt**, będące bazami zasobów z których dane czerpie program.
- **dialogs**, zawierający pliki od **dialog_0.txt** do **dialog_6.txt**, oraz **katalog_kowal.txt**, **katalog_sprzedawca.txt**, **katalog_zbrojmistrz.txt**, które zawierają tekst wyświetlany podczas dialogów z postaciami niezależnymi oraz handlu z nimi.

3.3 Ekspansywność projektu

Głównym zamysłem projektu było aby był ekspansywny, aby bez zbyt dużej ingerencji w pliki gry, można było dodać nowe rzeczy, czy to nowe przedmioty dla gracza, umiejętności, postacie, dialogi itp. Sprawilo to że zamiast wpisywać wszystkie dane bezpośrednio do programu zdecydowałem się skorzystać z zewnętrznych plików, z których następnie dane dodawane są do odpowiadających im wektorów, sygnowanych oznaczeniem BZ, będących skrótem od baz zasobów.

Stworzyło to swego rodzaju system danych przypominający bazę danych, w którym każdy element wpisany do bazy posiada swój

identyfikator oraz stałą ilość parametrów. Dzięki temu wypisywanie z bazy danych odbywa się dzięki funkcji, która pobiera nazwę bazy, id przedmiotu oraz index parametru, a następnie zwraca odpowiednią „linijkę” danych odpowiadającą następującemu wzorowi:

Zwracana_wartość=(id*ilość_parametrów)+index_parametru,

gdzie **ilość_parametrów** jest stała dla każdej z baz zasobów. Mimo że sam taki system wypisywania danych działa prawidłowo, zdaję sobie sprawę, że jest tutaj miejsce na poprawę oraz wprowadzenie lepszego sposobu przechowywania i odczytywania danych, np. z wykorzystaniem faktycznych baz danych.

4. Instrukcja użytkownika

Komendy

„Szkarałat” do swojego działania wykorzystuje komendy, które użytkownik wpisuje w konsoli, a następnie są one należycie interpretowane przez grę.

Podstawowymi komendami są:

- **pomoc** - wyświetla wszystkie dostępne w grze komendy.
- **N, E, S, W** – komendy będące oznaczeniami kierunków świata, służą do poruszania się w danym kierunku, o ile lokacja w której znajduje się gracz na to pozwala.
- **ekwipunek** – wyświetla pełny ekwipunek gracza, wraz z posiadaną ilością złota, wyekwipowaną bronią i zbroją.
- **umiejetnosci** - wyświetla listę umiejętności poznanych przez gracza.
- **statystyki** – wyświetla klasę gracza, imię, poziom doświadczenia, ilość punktów doświadczenia, ilość złota oraz aktualny poziom reputacji
- **spojrz** – wyświetla pełny opis lokacji składający się z:
 - Nazwy lokacji
 - Możliwe kierunki poruszania się
 - Opis lokacji
 - Kto znajduje się w tej lokacji
- **rozmawiaj** – (jeśli w danej lokacji znajduje się postać niezależna) rozpoczyna rozmowę z postacią niezależną.

- **zapisz** – zapisuje wszystkie statystyki, aktualną lokację oraz stan gracza do odpowiedniego pliku.
- **wyjdz** – służy do zakończenia rozgrywki, przed wyjściem gracz ma również opcję zapisu stanu gry.
- **handluj** - (jeśli w danej lokacji znajduje się postać niezależna, z którą można coś handlować) wypisuje możliwe do kupna przedmioty, użytkownik podaje liczbę znajdującą się obok przedmiotu który chce kupić i jeśli gracz posiada odpowiednią ilość złota to ten dany przedmiot zostaje dodany do ekwipunku gracza.
- **wymien** – (jeśli w danej lokacji znajduje się postać niezależna, u której można wymienić trofea) wyszukuje w ekwipunku przedmioty, które mają typ „trofea”, usuwa je i zwraca graczowi ich równowartość w złocie.
- **obejrzyj pancerz** – wypisuje nazwę, wartość zapewnianego pancerza oraz opis pancerza noszonego przez gracza.
- **obejrzyj bron** - wypisuje nazwę, wartość obrażeń oraz opis broni noszonej przez gracza.
- **obejrzyj przedmiot** – wypisuje przedmioty z ekwipunku użytkownika, ten podaje cyfrę przy przedmiocie który chce obejrzeć, a następnie wypisuje nazwę, typ, wartość oraz opis tego danego przedmiotu.
- **użyj przedmiot** – (dla przedmiotów typu „leczenie” i „szkarłat”) wypisuje przedmioty z ekwipunku użytkownika, podaje cyfrę przy przedmiocie który chce użyć, a następnie zależnie od typu zwiększa daną statystykę użytkownika.
- **wyrzuc przedmiot** - wypisuje przedmioty z ekwipunku użytkownika, ten podaje cyfrę przy przedmiocie który chce wyrzucić i usuwa go.
- **opis umiejetnosci** – wypisuje umiejętności użytkownika, ten podaje cyfrę przy umiejętności, której chce opis poznać, następnie wypisuje nazwę, koszt szkarłatu, obrażenia oraz opisu umiejętności.
- **przyjrzyj się** - (jeśli w danej lokacji znajduje się postać niezależna) wypisuje opis wyglądu postaci niezależnej.

System walki

Ważnym aspektem „Szkarłatu” jest turowy system walki, w którym użytkownik ma do wyboru jedną z trzech opcji:

1. atak – Gracz zadaje przeciwnikowi obrażenia, równe obrażeniom jego broni
2. Umiejętności – Gracz wybiera jedną z poznanych umiejętności i o ile posiada on wystarczającą ilość szkarłatu, używa jej zadając obrażenia i tracąc daną ilość punktów szkarłatu.
3. Przedmioty – Gracz wybiera jeden z posiadanych przedmiotów i o ile można go użyć, traci go oraz zyskuje wzrost danej statystyki, którą ten przedmiot zapewnia

Po ruchu gracza odbywa się ruch przeciwnika, który posiada 50% szans, że jego atak trafi gracza zadając obrażenia. Walka kończy się gdy ilość życia któregoś z uczestników dojdzie do 0. Jeśli gracz wygra walkę dostaje on punkty doświadczenia, szkarłat, reputację oraz trofeum po przeciwniku, natomiast jeśli przegra to gra się kończy, a gracz będzie musiał kontynuować grę od poprzedniego zapisu.

5. Podsumowanie i wnioski.

Udało mi się zrealizować wszystkie narzucone przeze mnie założenia co do projektu, jednakże wiem że można poprawić kilka jego aspektów. Z rzeczy które bym zmodyfikował albo całkowicie zmienił, byłby to cały system baz zasobów, które stworzyłem. Moim zdaniem istnieje możliwość, żeby go usprawnić, aby przechowywane dane nie musiały być składowane na zasadzie, jedna osobna wartość = jedna linia pliku.

Ponadto w pełni zmieniłbym system walki, z którego nie jest w pełni zadowolony. Zamiast określonych pól na których gracz ma szansę na rozpoczęcie walki z przeciwnikiem, wolałbym stworzyć system dzięki któremu gracz mógłby stoczyć walkę z jakąkolwiek postacią niezależną, przez co pokonanie jej mogłoby uniemożliwiać wykonanie poszczególnych zadań lub blokować możliwość kupna danych przedmiotów, przez co akcje gracza miałyby faktyczne konsekwencje dla jego gry.

Dodatkowo, w celu dalszego rozwoju programu można dodawać nową zawartość oraz powiększyć grywalny świat „Szkarałatu”, dodając większą liczbę lokacji, opisów oraz wydarzeń, aby wykreować jeszcze większy i ciekawszy świat, który zachęcałby graczy do dalszej rozgrywki.