Politechnika Śląska

Wydział Automatyki, Elektroniki i Informatyki

Programowanie Komputerów

Statki

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **autor** | **prowadzący** | **rok akademicki** | **kierunek** | **rodzaj studiów** | **semestr** | **termin**  **laboratorium** | **sekcja** | **Termin oddania sprawozdania** |
| Adam Warzecha | mgr inż. Grzegorz Kwiatkowski | 2020/2021 | informatyka | SSI | 3 | czwartek, 11:15-12:45  piątek,  8:15-9:45 | 2.2 | 03.11.2020 r. |

Link do GitHub: https://github.com/polsl-aei-pk3/195abe22-gr22-repo

**1. Analiza zadania**

Program pobiera dane podane naprzemiennie od użytkowników przez cały czas trwania rozgrywki. Podane dane są przetwarzane. Po zakończeniu rozgrywki wyniki są zapisywane do plików tekstowych.

**1.1 Struktury danych**

Struktury wykorzystane w programie to: Lista jednokierunkowa – dla przechowywania kolejnych elementów. Każdy element symbolizuje jedno pole na mapie.

**1.2 Kluczowe metody**

Jedną z najbardziej kluczowych metod w programie jest metoda typu void klasy **Mapa** zapiszMape(wygladMap\*& stat). Metoda przyjmuje jako parametr orginał początku listy jednokierunkowej. Następnie zapisuje do listy całą mapę, wraz z statkami gracza. Następną ważną metodą jest metoda typu void zapiszDoPlikuGracza1(fstream& plik). Metoda klasy **Gracz** przyjmuje jako parametr plik, do którego zostanie zapisana mapa statków, mapa strzałów oraz punktacja. Dla drugiego gracza metoda jest analogiczna.

**2. Specyfikacja zewnętrzna**

Program uruchomiony jest przez lokalny Debuger w programie Visual Studio. Aby rozpocząć rozgrywkę należy wprowadzić odpowiedni rozmiar.

**3. Specyfikacja wewnętrzna**

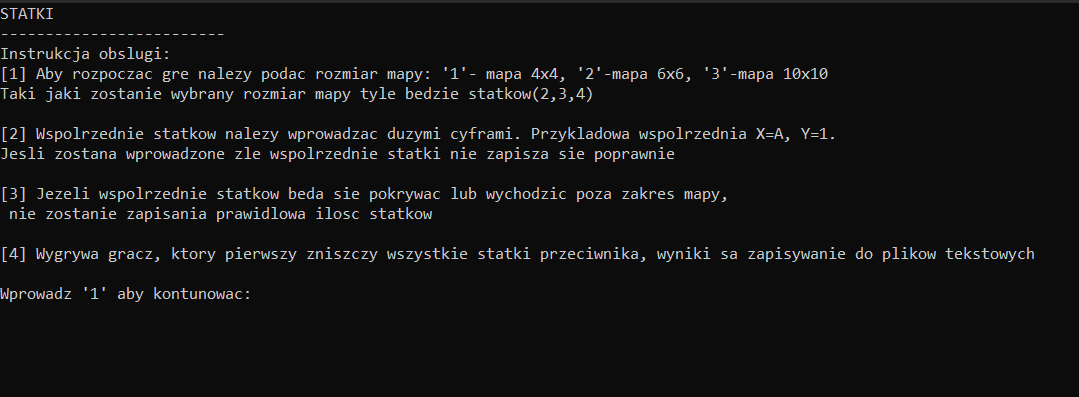
Program został zrealizowany zgodnie z paradygmatem obiektowym oraz posiada jedną dynamiczną strukturę danych. Program opiera się głównie na komunikacji z użytkownikiem. W programie rozdzielono interfejs (komunikację z użytkownikiem) od logiki aplikacji (tworzenie odpowiednio sformatowanych map oraz plików).

**4. Testowanie**

Program został przetestowany na różnego rodzaju danych wejściowych. Wprowadzanie błędnych statków nie powoduję błędów, jednakże owe statki nie zostaną uwzględnione na mapie. Użytkownik może wtedy rozpocząć rozgrywkę od nowa i podać poprawne dane. Program został sprawdzony pod kątem wycieków pamięci.

**5. Instrukcja obsługi**

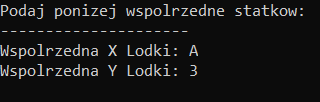
a) Na początku gracze powinni się zapoznać z uproszczoną instrukcją, która zostaje wyświetlona po uruchomieniu programu oraz wprowadzić ‘1’ aby przejść dalej.

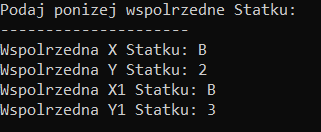


b) Następnie gracze muszą podać na jak dużej mapie rozgrywka ma się odbywać.

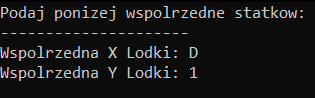


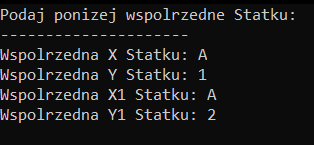
c) Na powyższym zdjęciu został wybrany rozmiar 1, czyli rozgrywka będzie się na mapie 4x4 a statkami tutaj będą łódka oraz statek. Pierwszy gracz wprowadza współrzędne dla łódki i następnie dla statku.



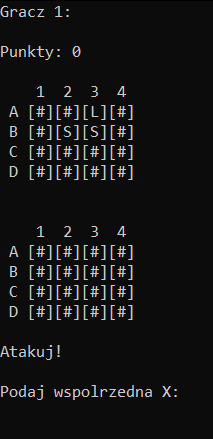


Analogiczne gracz drugi wprowadza swoje statki, pierwszy gracz powinien się odwrócić.

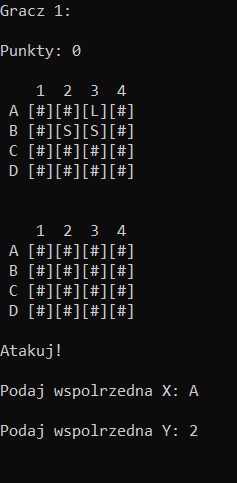




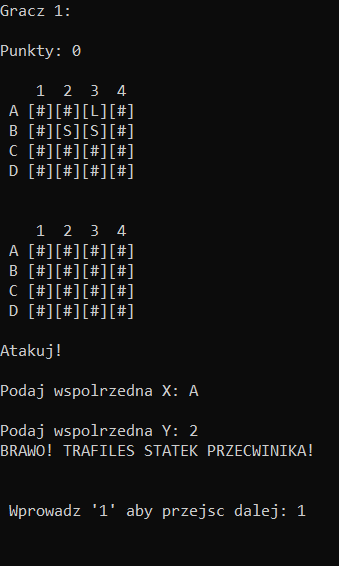
d) Po wprowadzeniu współrzędnych przez graczy rozpoczyna się właściwa rozgrywka, czyli strzały. Zaczyna gracz, który pierwszy wprowadzał statki. Na ekranie zostają wypisane 2 mapy. Pierwsza mapa to mapa statków gracza oraz strzałów przeciwnika. Druga mapa to mapa strzałów gracza.



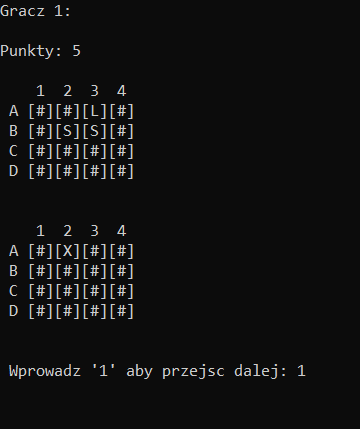
e) gracz atakuje po przez wprowadzenie współrzędnych na jakie chce oddać strzał.



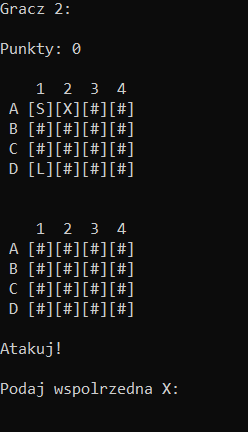
f) Po zatwierdzeniu zostaje wpisany komunikat o powodzeniu trafienia. Należy wprowadzić ‘1’ aby przejść dalej.



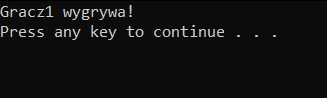
g) Następuje aktualizacja mapy. Aby rozpocząć ruch drugiego gracza należy wprowadzić i zatwierdzić ‘1’.



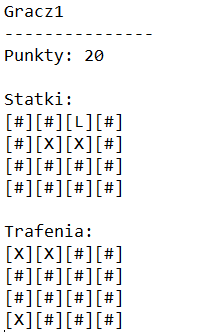
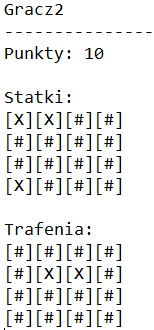
h) Wygląd map dla gracza drugiego. Kawałek statku został zatopiony. Gracz drugi oddaje strzał analogicznie do gracza pierwszego. Rozgrywka toczy się, aż do zatopienia wszystkich statków.



i) Gdy jeden z graczy zatopi wszystkie statki przeciwnika zostaje wypisany komunikat (w tym przypadku gracz 1).



j) Wyniki zostają zapisane do plików dla gracza pierwszego oraz drugiego.

**6. Wnioski**

Napisanie tego programu oraz nauczenie się rozumowania obiektowego w zaledwie ponad miesiąc było lekkim wyzwaniem. Program działa i spełnia wszystkie podanie elementy. Uważam, że gdyby było więcej czasu na napisanie projektu to udało by mi się go bardziej rozbudować np. dodać bibliotekę graficzną. Z uwagi, że jeszcze nigdy nie miałem styczności z bibliotekami graficznymi postanowiłem zrealizować projekt w konsoli Visuala. Największym problemem było sensownie dodanie operatorów do programu i ich wykorzystanie, ponieważ na początku nie widziałem sensu ich użycia. Ostatecznie to zagadnienie zostało zrealizowane i wykorzystane w kodzie.