

Politechnika Śląska
Wydział Matematyk Stosowanej
Kierunek Informatyka

Gliwice, 01.10.2020

Programowanie I
Projekt zaliczeniowy
"Chinczyk"

Konrad Kisiała gr. lab. 3/5

1. Opis projektu.

Chińczyk to gra dla od dwóch do czterech osób, w której każdy gracz wybiera 4 pionki jednego koloru, następnie kolejno rzucają kostką i przesuwają pionki w kierunku mety. Celem gry jest umieszczenie pionka jak najszybciej na linii mety. Wygrywa ten, który zrobi to jako pierwszy. Uwaga! Gracze mogą zbijać pionki, aby uniemożliwić przeciwnikom wygraną!

2. Wymagania

- Menu nawigacyjne po programie
- Wybór liczby graczy (2-4)
- Wpisywanie nicków graczy
- Podgląd planszy z aktualnym rozmieszczeniem pionów po każdym rzucie
- Możliwość wyboru pionka, którym chcemy wykonać ruch
- Kończenie rozgrywki w momencie ustawienia pionka na mecie
- Wyświetlanie kolorów w konsoli, by ułatwić rozgrywkę
- Efekty dźwiękowe
- Historia wyników (informacja o wygranym i jego przeciwnikach)
- Zasady gry w menu

3. Przebieg realizacji

Podczas przebiegu pisania takie rzeczy jak tworzenie planszy okazały się znacznie trudniejszy niż się tego spodziewałam na początku. Każde pole planszy musiało posiadać co najmniej swoje współrzędne x i y . Początkowe plany aby przechowywać to w zmiennych tablicowych okazały się niezbyt wygodne i wykonalne. Wybór padł na struktury. Problem pojawił się w tym jak nie pisać 40 struktur ręcznie. Z pomocą przyszło połączenie tych dwóch pomysłów.

```

// Tablica zawierająca wzor współrzędnych pol po której poruszają się pionki
int wszystkiePola[][2] = {
    {3,0},{3,1},{3,2},{3,3},
    {2,3},{1,3},{0,3},
    {0,4},{0,5},{0,6},{0,7},
    {1,7},{2,7},{3,7},
    {3,8},{3,9},{3,10},
    {4,10},{5,10},{6,10},{7,10},
    {7,9},{7,8},{7,7},
    {8,7},{9,7},{10,7},
    {10,6},{10,5},{10,4},{10,3},
    {9,3},{8,3},{7,3},
    {7,2},{7,1},{7,0},
    {6,0},{5,0},{4,0}
};

//Tworzenie 40 struktur (tyle ile pol) i przypisanie im współrzędnych ze wzoru
for (int i = 0; i < 40; i++) {
    pole[i].y = wszystkiePola[i][0];
    pole[i].x = wszystkiePola[i][1];
}

```

Użyte biblioteki:

iostream - standardowy strumień wejścia/wyjścia

string - obsługa zmiennych łańcuchowych (tekstowych)

- `empty()` sprawdza czy zmienna jest pusta

ctime - pozyskiwanie i manipulowanie datą i czasem

- `time(nullptr)` zwraca czas w milisekundach od 1 stycznia 1970, w projekcie służy jako ziarno do losowania liczb pseudolosowych

random - wprowadza możliwość generowania liczb pseudolosowych

- w projekcie użyty jest algorytm Mersenne Twister 19937

windows.h - zbiór różnych funkcji WinApi32

- `PlaySound()` odtwarza dźwięki w formacie *.wmv

fstream - strumień wejścia/wyjścia działający na plikach

- `open()` otwiera plik
- `is_open()` sprawdza czy plik jest otwarty
- `close()` zamyka plik

- << strumień zapisu do pliku

4. Instrukcja użytkownika

Zasady gry:

Aby wyjść pionkiem z bazy musisz wylosować liczbę 6 i wybrać pionek, pojawi się on na polu startowym.

Twoim zadaniem jest jak najszybciej przekroczyć jakimkolwiek pionkiem linie mety.

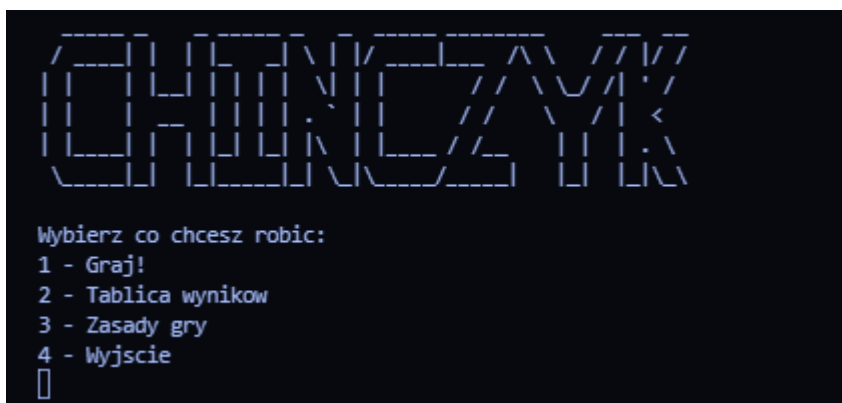
W trakcie gry można zbijać pionki innych graczy! Zbity pionek wraca wtedy do bazy.

Jeśli spróbujesz wykonać niedozwolony ruch (zabicie samego siebie) też wrócisz do bazy!

Uwaga! Wyjątkiem jest pole startowe, możesz tam umieścić kilka pionków jednocześnie.

Instrukcja obsługi:

Po uruchomieniu programu pokaże się baner i menu, z którego można wybrać różne opcje.



Między opcjami można dowolnie nawigować i wracać do menu głównego.

```
Wybierz co chcesz robic:
1 - Graj!
2 - Tablica wynikow
3 - Zasady gry
4 - Wyjście
2

Historia wynikow:

Wygral: Michal, przeciwnikami byli: Bartek Szymon
Wygral: Antek, przeciwnikami byli: Maciek Mateusz Pawel
Wygral: Konrad, przeciwnikami byli: Filip
Wygral: Bartosz, przeciwnikami byli: Kuba Boguslaw

Wcisnij 'x' aby wrocic do menu.
[]
```

Po wybraniu opcji graj, użytkownik zostanie zapytany o ilość graczy oraz ich nazwy

```
Wybierz co chcesz robic:
1 - Graj!
2 - Tablica wynikow
3 - Zasady gry
4 - Wyjście
1

Ilu jest graczy? (od 2 do 4)
3

Wpisz nazwe gracza nr 1
Marek

Wpisz nazwe gracza nr 2
Maciek

Wpisz nazwe gracza nr 3
Mateusz
```

Następnie nastąpi rozgrywka, program po każdym ruchu będzie wyświetlał mapę oraz informuje czyja jest teraz kolejka, wyświetli losowanie kostką oraz zapyta, którym pionkiem chcemy ruszyć

```

1 2   0 0 0 0 0 0   1 2
4 3   0   0   0   0   4 3
      0   0   0
0 0 0 0   0   0 0 0 0
0           0           0
0 0 0 0 0 0 0 0 0 0 0 0
0           0           0
0 0 0 0   0   0 0 0 0
      0   0   0
0 0   0   0   0   1 2
0 0   0 0 0 0 0   4 3

```

W przypadku wykonania niedozwolonego ruchu program poinformuje o tym

```

0 0 0 0 0 0 1 2
4 3 0 0 0 4 3
    0 0
1 2 0 0 0 0 0 0
0      0      0
0 0 0 0 0 0 0 0
0      0      0
0 0 0 0 0 0 0 0
    0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

```

W przypadku wygranej również :)



5. Podsumowanie i wnioski.

Prawie wszystkie założenia udało się zrealizować. Niektóre z nich takie jak tabela wyników, została zamieniona w historię wyników. Zostało to głównie podyktowane nie wyrobieniem się w terminie oddawania prac. Niektóre zasady gry zostały nieco zmienione lub uproszczone z tego samego powodu. Z różnic mojej wersji gry w Chińczyka a jej pierwotnej wersji mogę wymienić: wygrywa ten kto pierwszy przejdzie całą planszę zamiast uzupełnienia całej bazy, w mojej wersji jeśli gracz wykona niedozwolony ruch (zabicie samego siebie) to również jest odsyłany do bazy zamiast braku możliwości wykonania takiego ruchu.

Początkowo pisałem program równolegle na dwóch systemach - Windows i Linux z myślą aby gra była cross-platformowa jednak z przyczyn że nie wszystkie biblioteki są dostępne na obu systemach (np. windows.h) i brak jakiegoś przystępnego odpowiednika wymusiło pisanie tylko pod Windowsa.

Podsumowując spodziewałem się, że będzie to dużo prostszy projekt. W międzyczasie okazywało, że mechanizmy działania są bardziej skomplikowane. Jednak ostatecznie jestem zadowolony z efektów, gdyż był to mój pierwszy duży projekt.