

Snake

Diego R. Garzaro - GRR20172364

Visão Geral

Snake, também conhecido como “Jogo da Cobrinha” teve sua primeira versão em 1976, através do jogo “Blockade”. No fim dos anos 90 se tornou bastante popular em celulares da Nokia.

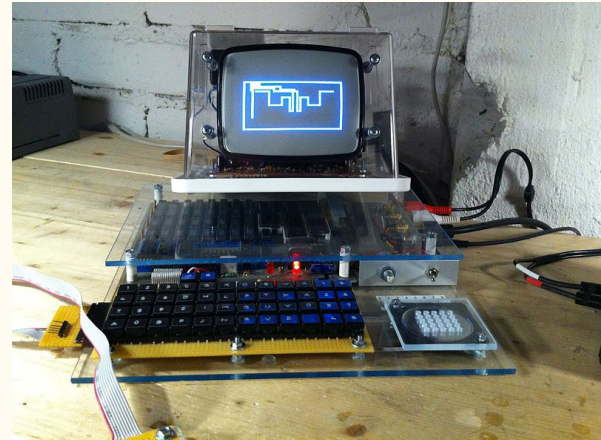


Fig. 1 - Snake em um Telmac 1800
Fonte: Wikipédia, 1978

Versões Snake

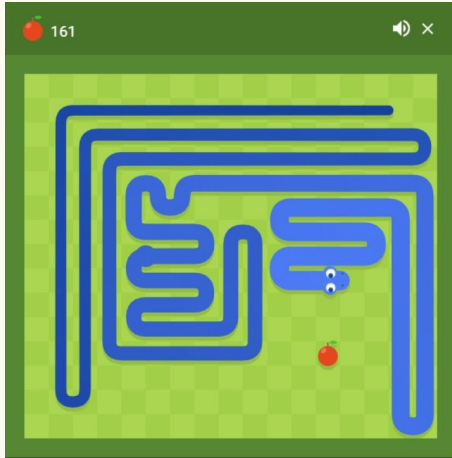


Fig. 2 - Google Play Snake
Fonte: YouTube, 2019



Fig. 3 - Nokia Snake
Fonte: India TV, 2018

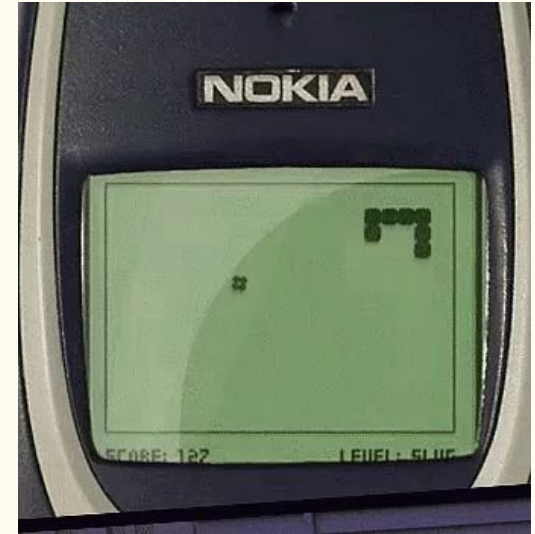


Fig. 4 - Nokia Snake
Fonte: Tenor, 2018

Desenvolvimento do Jogo

- Programação Orientada a Objetos;
- Sobrecarga de Operadores;
- Herança Simples;
- Herança Múltipla;
- Função Virtual;
- Polimorfismo;
- Interface Amigável;
- Diagrama de Classes.

Desenvolvimento do Jogo

Desafios:



Entrada do teclado;



Controlar tempo de movimento da serpente;



Atualizar elementos no console.

Desafios

➡ Entrada do teclado:

- “Conio.h” vs. “Windows.h”

MS-DOS

Windows API

Instável

Estável

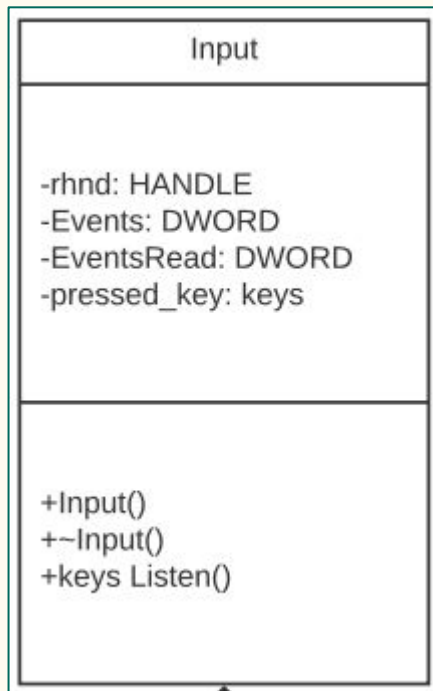


Fig. 5 - Diagrama *Input*

Desafios



Controlar tempo de movimento da serpente:

- Sleep(*DWORD dwMilliseconds*) vs. stopwatch(*int timeout*)

Suspende a execução da thread

Mantém a thread em execução

```
void Game::stopwatch(int timeout)
{
    if (clock() >= this->time)
    {
        this->time = clock() + timeout;
        this->update_game = TRUE;
    }
}
```

Fig. 6 - Código *stopwatch*

Desafios



Atualizar elementos no console:

- `system("cls")` vs. `set_cursor_position(int x, int y)`

Limpa o console

Sobrescreve o console

Atualiza o console

Move o cursor pelo console

```
void Board::set_cursor_position(int x, int y)
{
    static const HANDLE hOut = GetStdHandle(STD_OUTPUT_HANDLE);
    std::cout.flush();
    COORD coord = {(SHORT)x, (SHORT)y};
    SetConsoleCursorPosition(hOut, coord);
}
```

Fig. 7 - Código `set_cursor_position`

Diagrama de Classes

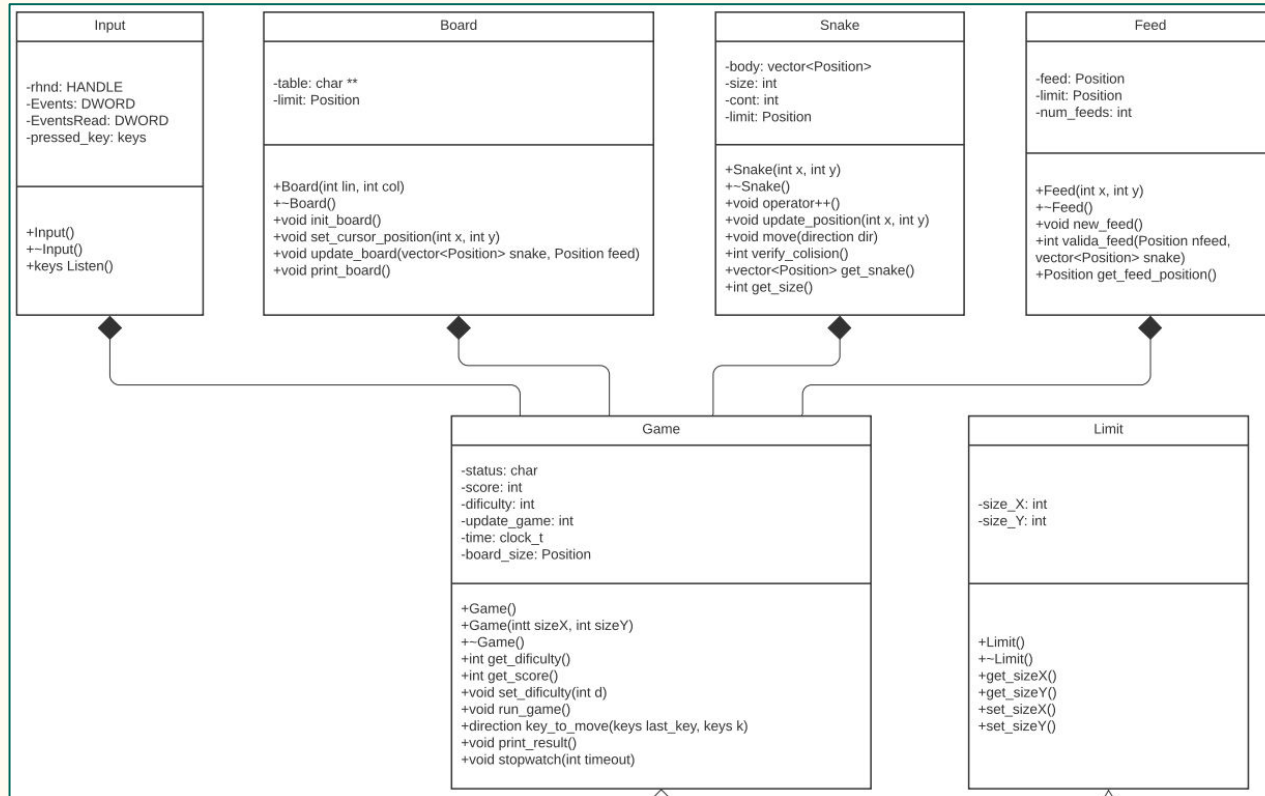


Fig. 8 - Diagrama de Classes pt.1

Diagrama de Classes

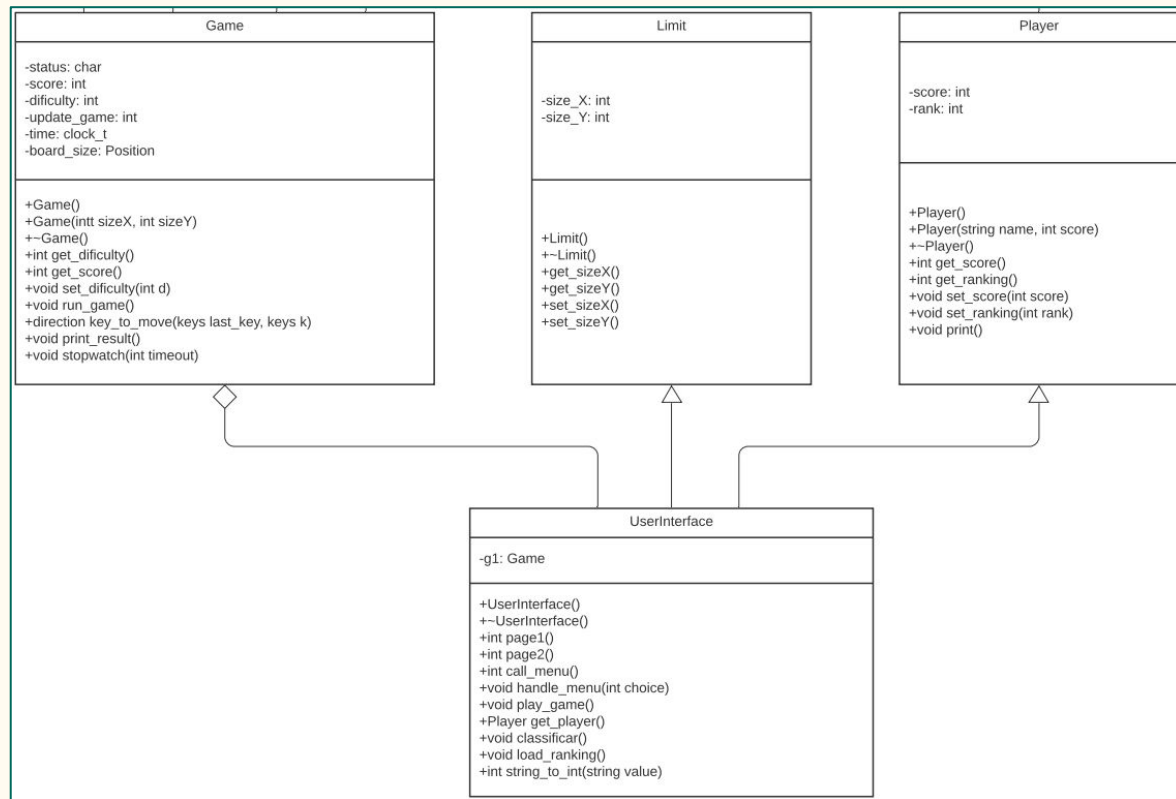


Fig. 9 - Diagrama de Classes pt.2

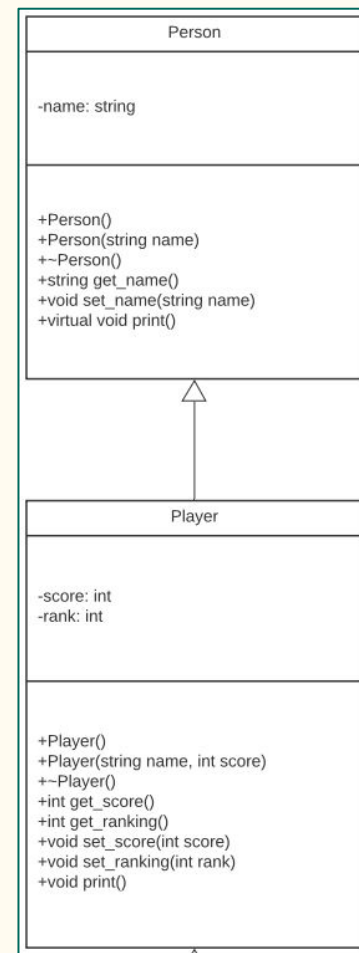


Fig. 10 - Diagrama de Classes pt.3

Diagrama de Classes

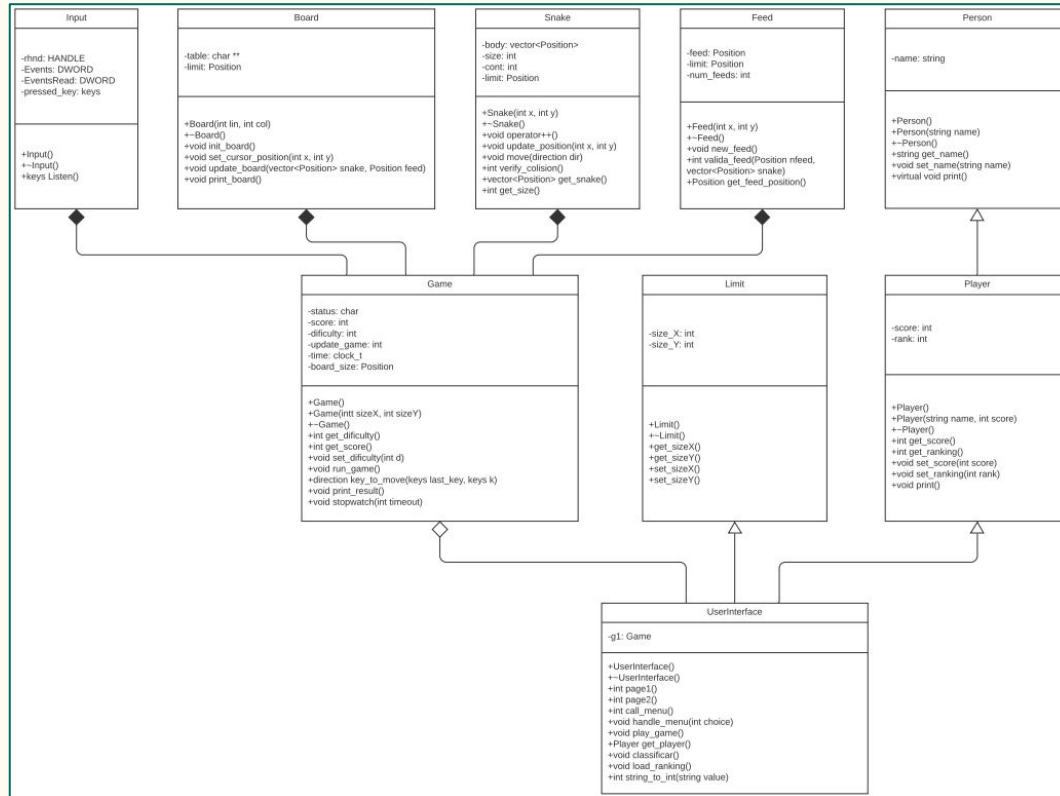


Fig. 11 - Diagrama de Classes Completo

Desenvolvimento do Jogo

Requisitos de projeto:

- ✓ Programação Orientada a Objetos;
- ✓ Sobrecarga de Operadores;
- ✓ Herança Simples;
- ✓ Herança Múltipla;
- ✓ Função Virtual;
- ✓ Polimorfismo;
- ✓ Interface Amigável;
- ✓ Diagrama de Classes.

```
Player: P00  
Score: 100_
```

Fig. 14.b - Função Virtual e Polimorfismo

```
void Snake::operator++()  
{  
    cont++;  
}
```

Fig. 12 - Sobrecarga Operadores

```
void Person::print(){  
    cout << this->name;  
}
```

Fig. 13.a - Função Virtual

```
void Player::print(){  
    cout << this->score;  
}
```

Fig. 13.b - Função Virtual

```
void UserInterface::classify()  
{  
    int is_qualified;  
    if (this->gl.get_difficulty() == 4)  
    {  
        Player p1 = get_player();  
        Person *p = &p1;  
        system("cls");  
        cout << "\n\n\t\t";  
        cout << "Player: " << p1.get_name() << endl  
            << "\t\tScore: ";  
        p->print();  
        Sleep(2000);  
        char file_name[15] = "ranking.txt";  
        this->load_ranking(p1, file_name);  
    }  
    else  
    {  
        return;  
    }  
}
```

Fig. 14.a - Função Virtual e Polimorfismo

Desenvolvimento do Jogo

Requisitos de projeto:

- ✓ Programação Orientada a Objetos;
- ✓ Sobrecarga de Operadores;
- ✓ Herança Simples;
- ✓ Herança Múltipla;
- ✓ Função Virtual;
- ✓ Polimorfismo;
- ✓ Interface Amigável;
- ✓ Diagrama de Classes.



Fig. 15 - Interface de Usuário 1

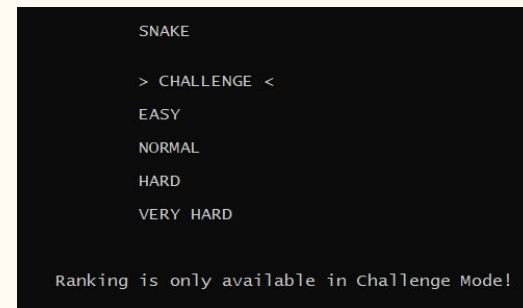


Fig. 16 - Interface de Usuário 2

Modo Challenge

- Mudança da dificuldade do jogo de acordo com a pontuação;

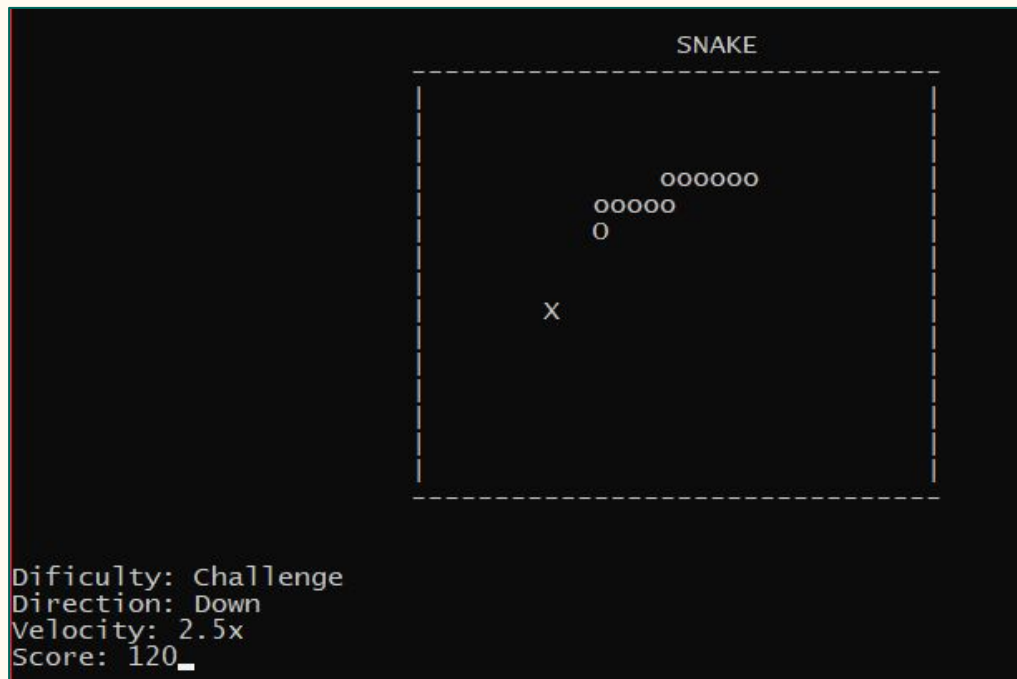


Fig. 17 - Snake no modo Challenge

Modo Challenge

- Sistema de qualificação dos melhores jogadores (Ranking).



1.	Die	460
2.	DDD	310
3.	DRG	260
4.	AAA	210
5.	tes	140
6.	Tes	120
7.	POO	100
8.	TES	80
9.	tes	60
10.	Die	40

Press any key to continue . . . _

Fig. 18 - Ranking do modo Challenge

Demonstração do Jogo



Snake

Obrigado!

Diagrama de Classes e código fonte do jogo estão disponíveis no GitHub:

Link: <https://github.com/DiegoGarzaro/POO/tree/main/TrabalhoFinal>

Diego R. Garzaro - GRR20172364