

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Úvod do počítačových sítí

Lodě

A13B0481P

Adam Barák

barak.adam94@gmail.com

Obsah

1	Programátorská dokumentace	1
1.1	Řešení	1
1.2	Server	1
1.2.1	Datové struktury	1
1.3	Klient	3
1.3.1	Datové struktury	3
1.4	Formáty vyměňovaných zpráv	3
1.5	Použité programy	4
2	Uživatelská dokumentace	6
2.1	Požadavky a přeložení	6
2.2	Spuštění, parametry programu, ovládání	6

1 Programátorská dokumentace

1.1 Řešení

Aplikace se skládá z klienta a serveru. Komunikace je uskutečněná přes TCP. Server je naprogramovaný v jazyce C a má za úkol řešit jakoukolik komunikaci mezi klienty a spravovat herní pole každého hráče. To znamená, že server rozhoduje, když někdo prohraje, když je potopena loď, nebo naopak, když hráč nezasáhne nic.

Server

Při spuštění serveru se zahájí přijímání příchozích připojení od klientů. Vždy poté, co se připojí dva klienti se pomocí příkazu fork vytvoří nový proces pro hru a nová vlákna pro naslouchání klientům. Poté je přijata zpráva o přezdívce a rozložení lodí od obou zúčastněných hráčů. Tím hra začíná. Poté server přijímá zprávy o útocích a odesílá bitevní výsledky. Pokud se některý z klientů v průběhu hry odpojí, nebo hra dojde ke konci, server odešle klientům příslušné zprávy a ukončí hru.

Klient

Klient po spuštění požádá uživatele o zadání přezdívky a adresy serveru. Poté se připojí k serveru a čeká, až si uživatel umístí lodě do hracího plánu a poté je hra spuštěna. Oba hráči se střídají v tazích až do konce hry, nebo do té doby než hráč klienta vypne, nebo přijde informace o odpojení ze serveru.

1.2 Server

1.2.1 Datové struktury

Server obsahuje pouze dvě datové struktury a to strukturu Game a strukturu Player

Game

Struktura game je struktura, jenž představuje jednu hru lodí. Struktura obsahuje vlastní ID, což značí identifikační číslo dané hry, pole dvou hráčů z nichž jeden má ID 0 a druhý 1. Informaci o ID je mimo pole také uchována přímo ve struktuře hráče.

```
struct game {  
    int ID;  
    int turn;  
    struct player player[2];  
};
```

Player

Struktura player je struktura hráče, která obsahuje veškeré důležité informace o hráči, díky kterým se může hra spojit s klientem, manipulovat s vláknem, jenž naslouchá klientovi, používat hráčovu přezdívku a v první řadě obsahuje hráčovo herní pole. Herní pole je tvořeno dvojrozměrným polem čísel o velikosti 15 x 15, v němž každá z lodí má své jedinečné ID, kterým je symbolizována. Pokud je loď zasáhnutá, buňka v poli nabyde hodnotu 0 a prozkoumáme pole, zdali se v něm vyskytuje ještě nějaká část zasažené lodě. Pokud ne, loď je vyhodnocena jako potopená. Pokud se v poli nevyskytuje už žádná loď, hráč prohrál.

```
struct player {  
    int id;  
    char nick[12];  
    int nick_length;  
    int socket; /*player address*/  
    int grid[15][15];  
    pthread_t reader;  
    struct game* parentGame;  
};
```

1.3 Klient

1.3.1 Datové struktury

Connection

Balík connection obsahuje 4 třídy pro navázání komunikace, udržování komunikace a kódování zpráv.

- Coder - pro odoování a dekódování zpráv
- Connection - třída pro připojení
- Reader - vlákno, které neustále čte příchozí data ze serveru
- Writer - pomocí writeru se odesílají data na server

Core

Balík core obsahuje hlavní třídy aplikace (jádro) a to třídy Main, Resources, Config a Info

Game

Balík game obsahuje struktury hracích mřížek pro spojence a nepřítel

GUI

Balík GUI obsahuje grafické rozhraní klienta

1.4 Formáty vyměňovaných zpráv

Server odesílá pouze 5 druhů zpráv a to:

prefix	
R	Opponent nickname
E	Optional postfix
D	Coordinates to defend
A	Attacked coordinates + Result

R - Zpráva o tom, že hra je připravena a jak se jmenuje protihráč

E - Hra je ukončena. Pokud po E nenásleduje žádný znak, znamená to, že se odpojil protihráč. Pokud po E následuje S, server ukončil komunikaci. Pokud je znakem N, nebylo ani možné se k serveru připojit. Pokud přišla zpráva, která začíná na E a pokračuje neznámým znakem, jedná se o chybové hlášení a aplikace je ukončena

D - obsahuje pouze souřadnice na které zaútočil nepřítel, popřípadě vyhodnocení, že jste prohrál a to v případě, že byla potopena vaše poslední loď a za souřadnicí se ještě vyskytuje navíc znak '3'

A - Odpověď na útok, který jsme provedli proti nepříteli, který pro ujistění opakuje souřadnice, na které bylo útočeno a výsledek, který je pro: 0 znamená, že jsme netrefili žádnou loď. Pro 1 znamená, že jsme loď trefili, ale nepotopily. Pokud 2, potopili jsme loď a 3 značí, že jsme potopili poslední loď soupeře

Klient odesílá pouze 4 druhy zpráv a to:

prefix			
L	Nick length	Nick	Grid
A	Coordinates		
X	Error		
E			

L - Zpráva o tom, že je hráč připraven začít hrát. Obsahuje délku hráčovi přezdívky, přezdívku a souřadnice středů jednotlivých lodí v mřížce ve tvaru velkých písmen

A - Zpráva o tom, že chceme zaútočit, obsahuje navíc souřadnice kam útočíme

X - Zpráva o tom, že poslední zpráva byla špatně formátována

E - Zpráva o tom, že klient ukončuje hru

1.5 Použité programy

Při vytváření aplikace byly použity tyto programy:

Dokumentace

- LaTeX - overleaf online editor

Repozitář

- GitHub repository

Server

- Bloodshed Dev-C++ - pro vývoj aplikace
- Bash on Ubuntu on Windows - pro spuštění aplikace

Klient

- Eclipse IDE - pro vývoj aplikace
- CorelDraw - vzhled GUI
- PhotoFiltre - dodatečné zpracování obrázků

2 Uživatelská dokumentace

2.1 Požadavky a přeložení

Server

Server funguje na operačním systému Linux

Přeložení serveru je možné pomocí příkazu:

```
gcc -pthread -o server main.c game.c player.c
```

Tím se vytvoří spustitelný soubor "server", který spustí server

Klient

Pro spuštění klienta je zapotřebí mít nainstalovanou Javu a to minimálně verzi 1.8.

2.2 Spuštění, parametry programu, ovládání

Pro klienta nepoužíváme žádné parametry. Při spuštění zadáme přezdívku, IP adresu serveru a port a poté se řídíme pokyny v nabídce nápověda. Server může být spuštěn s parametrem pro port serveru, pokud nebude, port bude nastaven na 6789. Další ovládání serveru není možné, je plně automatický.