

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA APLIKOVANÝCH VĚD

DOKUMENTACE K
SEMESTRÁLNÍ PRÁCI
KIV/UPS
SÍŤOVÁ HRA "SEDMA"

Martin Červenka A14B0239P
cervemar@students.zcu.cz

31. prosince 2016

Obsah

1	Zadání	2
2	Pravidla hry	3
2.1	Karty	3
2.2	Cíl hry	3
2.3	Rozdávání	3
2.4	Průběh hry	3
2.5	Bodování	3
3	Překlad systému	4
3.1	Automatická instalace	4
3.2	Překlad serveru	4
3.3	Překlad klienta	4
4	Spuštění aplikace	5
4.1	Spuštění pomocí skriptu	5
4.2	Spuštění serveru	5
4.3	Spuštění klienta	6
5	Programátorská dokumentace	7
5.1	Popis řešení	7
5.1.1	Server	7
5.1.2	Klient	7
5.2	Hlavní části aplikace	8
5.2.1	Server	8
5.2.2	Klient	8
5.3	Protokol	9
5.4	Žádosti	9
5.5	Odpovědi	10

1 Zadání

- Úlohu naprogramujte v programovacím jazyku C/C++ anebo Java. Pokud se jedná o úlohu server/klient, pak klient bude v Javě a server v C/C++.
- Komunikace bude realizována textovým nešifrovaným protokolem nad TCP protokolem.
- Výstupy serveru budou v alfanumerické podobě, klient může komunikovat i v grafice (není podmínkou).
- Server řešte pod operačním systémem Linux, klient může běžet pod OS Windows XP. Emulátory typu Cygwin nebudou podporovány.
- Realizujte konkurentní (paralelní) servery. Server musí být schopen obsluhovat požadavky více klientů souběžně.
- Součástí programu bude trasování komunikace, dovolující zachytit proces komunikace na úrovni aplikačního protokolu a zápis trasování do souboru.
- Zdrojové kódy organizujte tak, aby od sebe byly odděleny části volání komunikačních funkcí, které jste vytvořili na základě zadání, od částí určených k demonstraci funkčnosti vašeho řešení (grafické rozhraní).

2 Pravidla hry

Zjednodušený popis pravidel dle [1] je uveden v následujících subsekcích.

2.1 Karty

32listové, nejsilnějším listem hry jsou sedmy. Přebíjejí kteroukoliv kartu. Přitom zde barvy nehrají prakticky žádnou roli, stejně tak tu odpadá pořadí hodnot. Vyšší karta se proti nižšímu listu nijak neuplatní. Přebíjet může nanejvýš karta stejné hodnoty. Kluk zabijí kluka, osma osmu. Univerzální hodnotou jsou už zmíněné sedmy. Jediné karty, které přinášejí vítězství a přiměřené bodové ohodnocení jsou desítky a esa, tzv. hodnotné listy.

2.2 Cíl hry

Získat co nejvíce zdvihů obsahujících esa a desítky. Kromě toho získat body za poslední zdvih ve hře.

2.3 Rozdávání

Rozdá se po čtyřech listech, zbývající karty se jako talón položí na stůl. Odsud si hráči po každém zdvihu dokupují, aby doplnili stav listů v ruce na původní počet.

2.4 Průběh hry

Předák otevírá hru. Protože se vzájemně přebíjejí jen karty stejné hodnoty, je účelné vynést od "dvojáka", tj. kartou, kterou máme zakoupenou alespoň dvakrát. Barvu netřeba ctít, ale není zapotřebí ani přebíjet. Každý hráč je povinen odhodit kartu v libovolné barvě nebo libovolné hodnotě. Sedmy většinou držíme v ruce tak dlouho, dokud nedojde k boji o hodnotné listy – esa a desítky.

2.5 Bodování

Po skončení sehrávky spočítají partneři své desítky a esa. Každá z těchto karet představuje 10 bodů stejně jako poslední zdvih. Maximálně dosažitelný zisk má takto dohromady hodnotu 90 bodů. Vyhrává hráč, který po dlouhé sérii her získal nejvyšší bodový náskok.

3 Překlad systému

Pro to, abychom mohli spustit systém je nejprve důležité systém přeložit. V následujících podsekcích je postup tohoto překladu.

3.1 Automatická instalace

Program lze automaticky nainstalovat pomocí jednoho skriptu. Tento skript se nachází v kořenové složce projektu a jmenuje se **run**. Nepotřebujeme ani mít celý projekt stažený, protože se stáhne sám automaticky. K stažení celého projektu včetně závislostí pak stačí spustit příkaz **run** s parametrem **p**:

- `cd cesta_do_slozky_s_projektem`
- `./run p`

Po dokončení příkazu **run** máme celý projekt stažený a přeložený, můžeme tedy přejít do sekce Spuštění aplikace.

3.2 Překlad serveru

V případě, že z nějakého důvodu nelze automatikou instalaci použít, nebo instalace selže, je možné projekt snadno přeložit ručně.

Pro překlad serverové části systému je potřeba mít nainstalovaný program **CMake** a program **make**. Překlad poté provedeme následující sekvencí příkazů:

- `cd cesta_do_slozky_s_projektem`
- `mkdir build`
- `cd build`
- `cmake ..`
- `make`

Ve složce **build** vznikne přeložený program s názvem **server**.

3.3 Překlad klienta

Na překlad klienta potřebujeme javu v minimální verzi 8. Překlad pak provedeme sekvencí příkazů:

- `cd cesta_do_slozky_s_projektem/client/src`
- `javac cards/client/*.java -Xdiags:verbose -Xlint:unchecked -d ../bin/`

4 Spuštění aplikace

Po překladu aplikace je nyní možné aplikaci spustit. Postup pro spuštění jednotlivých částí systému je uveden níže.

4.1 Spuštění pomocí skriptu

Je možné jak server tak i klienta spustit pomocí skriptu **run** (viz sekce Překlad systému). Server se spustí s parametrem **s**, klient s parametrem **c**.

Příklad spuštění klientské části:

- `cd cesta_do_slozky_s_projektem`
- `./run c`

4.2 Spuštění serveru

Po překladu se program nachází ve složce `build`, proto do této složky přejdeme a pak spustíme server. Defaultní port pro aplikaci je 43158:

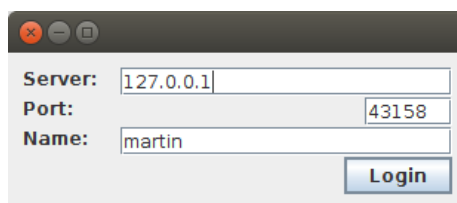
- `cd cesta_do_slozky_s_projektem/build`
- `./server 43158`

4.3 Spuštění klienta

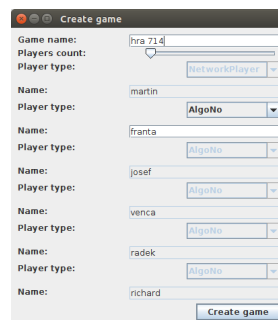
Přeložený klient se nachází ve složce **client/bin**. spuštění provedeme příkazy:

- `cd cesta_do_slozky_s_projektem_/client/bin`
- `java cards.client.Main`

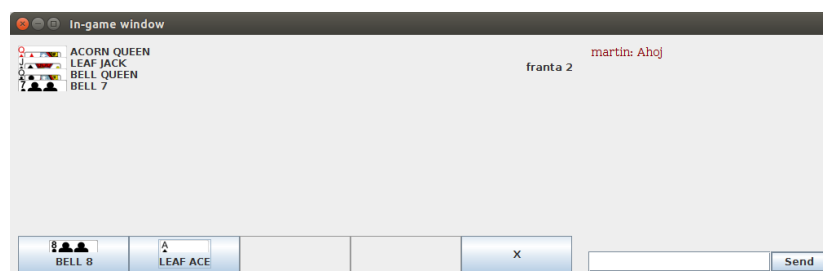
Po provedení příkazů se zobrazí okno podobné jako na obr. 1. Pokud známe IP adresu a port serveru, můžeme se připojit. Server ověří, zda-li existuje hra, kde by daný hráč mohl hrát. Pokud ano, zobrazí se okno hry (viz obr. 3, jinak se zobrazí okno vytvoření hry (viz obr. 2).



Obrázek 1: Přihlašovací okno



Obrázek 2: Okno vytvoření hry



Obrázek 3: Herní okno

5 Programátorská dokumentace

Jak je aplikace naprogramována a jak funguje je stručně shrnuto v této sekci. Nejprve (v sekci 5.1) je uveden slovní popis funkce programu, v sekci 5.2 je pak uvedeno kde lze tyto funkce v programu nalézt.

5.1 Popis řešení

Následuje stručný popis jakým způsobem aplikace pracuje. Popis je primárně zaměřen na komunikaci serveru a klienta, nepopisuje tedy jakým způsobem funguje GUI na klientské aplikaci nebo jak byla naprogramována hra samotná.

5.1.1 Server

Program je multivláknový, každé připojení má své vlastní naslouchací vlákno. Pokud přijde zpráva od klienta, server identifikuje zprávu (zprávy se identifikují podle jejich začátků) a pošle parametry zprávy do obslužné funkce. Tato obslužná funkce se provede a znovu nastane fáze naslouchání zpráv. Tato procedura se opakuje dokud nenastane nějaká chyba v komunikaci nebo pokud se klient softwarově neodpojí.

5.1.2 Klient

Stejně jako na serverové části aplikace je i klientská část multivláknová. Jedno vlákno se stará o grafické uživatelské rozhraní (GUI) a další o komunikaci se serverem. Aplikace pak pracuje na systému změtného volání (tzv. callbacku); vlákno starající se o GUI předá vláknu starajícímu se o komunikaci funkci kterou provede pokud přijde definovaná funkce.

5.2 Hlavní části aplikace

V následujících podsekcích je uvedeno kde lze nalézt důležité části kódu. To umožňuje rychlejší orientaci v kódu.

5.2.1 Server

Tabulka 1 popisuje kde se v kódu serverové části aplikace nacházejí hlavní funkční části. Vždy je uvedena vlastnost, pak název souboru a řádek v daném souboru kde začíná důležitá část kódu.

Funkce	Soubor	Řádka
Spuštění aplikace	main.cxx	14
Spuštění serveru	server.cxx	64
Vytvoření nového spojení	server.cxx	81
Naslouchání	server.cxx	124
Tržídění příchozích zpráv	commands.cxx	176
Zpracování příchozích zpráv	commands.cxx	186
Pole obslužných funkcí	commands.cxx	11
Odeslání zprávy	server.cxx	137
Příprava nové hry	game.cxx	108
Začátek nové hry	game.cxx	75

Tabulka 1: Tabulka hlavních částí programu serveru.

5.2.2 Klient

Důležité funkční části klientské části aplikace se nacházejí v tabulce 2. Vždy je uvedena vlastnost, pak název souboru a řádek v daném souboru kde začíná důležitá část kódu.

Funkce	Soubor	Řádka
Spuštění aplikace	Main.java	7
Spuštění klienta	Client.java	65
Vytvoření spojení	Client.java	77
Naslouchání	Client.java	85
Tržídění příchozích zpráv	Client.java	114
Zpracování příchozích zpráv	Client.java	40 a 122
Pole obslužných funkcí	Client.java	23
Odeslání zprávy	Client.java	53

Tabulka 2: Tabulka hlavních částí klientského programu.

5.3 Protokol

Server a klient musejí používat protokol, aby spolu mohli spolupracovat. V této sekci jsou popsány zprávy, které si mezi sebou posílají. V tabulce 3 se nacházejí společné části zpráv.

one_word	"a..zA..Z" { "a..zA..Z" }
card	("HEART" "LEAF" "BELL" "ACORN") "" ("7" "8" "9" "X" "JACK" "QUEEN" "KING" "ACE")
player_name	("ProgrammerBot" "AlgoNo" "NetworkPlayer" "Person")
game_params	["NumberOfPlayers="number] { "Name="one_word "Player="player_name }

Tabulka 3: Tabulka obecných položek

5.4 Žádosti

V této sekci jsou vypsané zprávy, které odesílá klient na server. Nejprve jsou uvedeny bezparametrické zprávy, které očekávají odpověď od serveru.

Pokud je pravá poloviční buňka tabulky červená a nachází se v ní nějaký odkaz, tak to znamená, že se jedná o reakci na danou událost, která od serveru přišla (viz sekce Odpovědi). Jinak se jedná o asynchronní zprávy, které nejsou nijak vynucovány serverem.

přivítání	
"WELCOME"	
odpojení	
"DISCONNECT"	
jaké mám karty	
"WHAT ARE MY CARDS"	
jaké jsou počty karet	
"WHAT ARE CARD COUNTS"	
jaké karty jsou na stole	
"WHICH CARDS ARE ON TABLE"	
přihlášení	
"LOGIN AS "one_word	
hrají	hraj
"PLAY "card	
vytvoření hry	
"CREATE GAME "oneword ""game_params	
poslání zprávy	
"SEND MESSAGE"oneword { ""oneword }	
volím	volba
"VOTE "("YES" "NO")	
ping	
"PING"	

Tabulka 4: Tabulka žádostí klienta k serveru

5.5 Odpovědi

V této sekci jsou uvedeny zprávy které přicházejí ke klientovi ze serveru. Jako v předchozí sekci pokud se v pravé poloviční buňce nachází nějaký text, jedná se o odkaz na zprávu, která přišla od klienta a vynucuje danou odpověď.

špatná zpráva	jakákoliv akce
"BAD COMMAND" { "" oneword }	
zahraná karta	
"CARD PLAYED" card one_word	
tvé karty jsou	jaké mám karty
"YOUR CARDS ARE" ["" card] ["" card] ["" card] ["" card]	
hra existuje	přihlášení
"GAME EXISTS"	
hra vytvořena	vytvoření hry
"GAME CREATED"	
hráč již existuje	přihlášení
"THIS PLAYER IS ALREADY CONNECTED"	
počty karet jsou	jaké jsou počty karet
"CARD COUNTS FOLLOWS" { "" one_word "0..4" }	
vítej	
"WELCOME"	
hraj	
"PLAY"	
zpráva od	poslání zprávy
"MESSAGE FROM "one_word ""one_word { ""one_word }	
rozloučení	odpojení
"BYE"	
volba	
"VOTE FOR REPEAT"	
na stole jsou	jaké karty jsou na stole
"ON TABLE IS" { "" card }	
výsledek hry	
"GAME RESULT IS " ("win" "draw" "lose")	
pong	ping
"PONG"	

Tabulka 5: Tabulka odpovědí serveru klientovi

Reference

- [1] Pravidla hry sedma:
Omasta V., Ravik S., Karty, hráči, karetní hry, 80-7309-206-9, str. 143-145