

Programovacie jazyky pre vstavané systémy

Semestrálna práca

Infinite Pong

Andrej Šišila

Používateľská príručka

“Infinite Pong” je sieťová hra pre dvoch hráčov. Úlohou hry je čo najviac krát odraziť loptičku od plošiny. Hráč skóruje, ak sa loptička odrazí od steny za protihráčovou plošinou. Vyhráva hráč, ktorý skóruje najviac bodov. Výsledok hry sa pri ukončení hry zobrazí na obrazovke.

Spustenie hry

Hra pozostáva zo serverovej a klientskej aplikácie. K serveru sa môže pripojiť najviac jeden klient – protihráč.

Aby sa mohli dvaja hráči hrať, musí najprv jeden z nich spustiť serverovú aplikáciu v termináli:

```
infinite_pong_server <port>
```

napríklad:

```
infinite_pong_server 8888
```

Potom druhý hráč spustí klientskú aplikáciu a pripojí sa k serveru:

```
infinite_pong_client  
<domenove_meno_servera_alebo_ip_adresa_servera> <port>
```

napríklad:

```
infinite_pong_client localhost 8888
```

alebo

```
infinite_pong_client 127.0.0.1 8888
```

Po vykonaní príkazu od klienta sa otvorí okno s hrou u oboch hráčov (klienta aj servera).

Ovládanie hry

Aj klientská, aj serverová aplikácia sa ovládajú rovnakým spôsobom:

W	pohyb plošiny nahor
S	pohyb plošiny nadol
Esc	ukončenie a vyhodnotenie hry

Po ukončení hry sa jej vyhodnotenie zobrazí v termináli a vo vrchnej časti hracej plochy namiesto skóre.

Programátorská príručka

Hra “Infinite Pong” je sieťová hra pre dvoch hráčov implementovaná v jazyku C. Hra využíva funkcionality knižnice “SDL2” na graficky akcelerované vykresľovanie prvkov a rozšírenie “SDL2_ttf” na vykresľovanie písma pomocou “SDL2” knižnice. Hra bola vyvíjaná na operačnom systéme “Debian 8.6.0 x64”.

Server je zodpovedný za počítanie súradníc herných prvkov, načítavanie vstupov od seba aj klienta a vykresľovanie stavu hry.

Klient je zodpovedný za načítavanie vlastných vstupov, ich následné odosielanie serveru a vykresľovanie stavu hry na základe informácií poskytnutých serverom. Klient v sebe neobsahuje žiadnu logiku hry – tá je uložená na serveri.

V adresári práce sa nachádza ešte súbor “digital-7 (mono).ttf”. Je to súbor, ktorý obsahuje typ písma, ktoré používam na vypisovanie skóre v hre.

Kompilácia programu

Kompilácia serverovej / klientskej aplikácie je možná cez priložený Makefile alebo priamo cez terminál.

Kompilácia cez Makefile:

```
make          # vytvorí aj klientskú, aj serverovú aplikáciu
make all      # to isté ako 'make'
make clean    # vymaže existujúce objektové a binárne súbory
make infinite_pong_server # vytvorí spustiteľný súbor servera
make infinite_pong_client  # vytvorí spustiteľný súbor klienta
```

Na kompiláciu oboch aplikácií stačí vykonať iba príkaz `make`. Pri rekompilácii oboch aplikácií je potrebné vykonať príkazy `make clean` a následne `make`.

Kompilácia servera cez terminál:

```
gcc -I "/usr/include/SDL2" -c infinite_pong_server.c -lSDL2
-lSDL2_ttf -o infinite_pong_server.o
gcc infinite_pong_server.o -lSDL2 -lSDL2_ttf -o infinite_pong_server
```

Požadované prvky v semestrálnej práci

- **Polia**

Väčšina polí je v aplikácií alokovaná v dynamickej pamäti. Každé takéto pole je dealokované funkciou `free()` a následne je adresa poľa nastavená na `NULL`.

- **Štruktúry**

Všetky štruktúry sú v aplikácií alokované v dynamickej pamäti. Každá takáto štruktúra je dealokovaná funkciou `free()` a následne je adresa štruktúry nastavená na `NULL`. Nižšie je uvedený zoznam vlastných štruktúr:

- **Player**
 - ukladá súradnice, výšku a šírku plošiny hráča
- **Ball**
 - ukladá súradnice loptičky a jej rýchlosť po osi x a y
- **Game**
 - ukladá šírku a výšku okna
- **Init**
 - správa posielaná po sieti od servera ku klientovi, ktorou server oznamuje inicializačné parametre hry klientovi.
- **Message**
 - správa posielaná po sieti striedavo medzi klientom a serverom, ktorou server / klient oznamuje informácie o stave hry klientovi / serveru.
- **Smerníková aritmetika**

Všetky jednoslovné reťazce v klientskej aj serverovej aplikácii sa vypisuje pomocou smerníkovej aritmetiky metódou “printToTerminal”. Tá berie ako parametre adresu reťazca a jeho dĺžku.
- **Vlákná**

Nie sú v použité v aplikácii.
- **Smerníky na funkcie**

Nie sú v použité v aplikácii.
- **Sockety**

Klientská aj serverová aplikácia používa TCP socket. Socket je inicializovaný a uzatváraný vo funkcii `main()`.

Komunikácia medzi klientom a serverom prebieha v nasledovných krokoch:

- 1) Prvý hráč spustí serverovú aplikáciu. Aplikácia začne sledovať prichádzajúce spojenia
- 2) Druhý hráč spustí klientskú aplikáciu.
- 3) Akonáhle sa klient pripojí na server, klient odošle serveru správu “start”.
- 4) Server prijíme správu “start” a následne naplní štruktúru “Init” údajmi potrebnými na inicializáciu hry u klienta. Po naplnení tejto štruktúry ju server odošle klientovi, ktorý si podľa nej nastaví parametre danej hry. Tým je zaistené, že klient aj server majú rovnako nastavené parametre hry t.j. hra je férová pre oboch hráčov.
- 5) Teraz klient aj server vôjdu do programového cyklu.
- 6) Najprv server naplní štruktúru “Message”, ktorá v sebe nesie informácie o aktuálnom stave hry, a odošle ju klientovi cez vytvorený socket.
- 7) Klient prijíme túto správu a naplní ňou svoju štruktúru “Message”. Klient zároveň skontroluje, či prijatá správa nemá nastavenú premennú na ukončenie hry. Ak áno, skočíme na krok 13).
- 8) Klientská aplikácia načíta vstup z klávesnice, zapíše ju do svojej štruktúry “Message” a odošle správu serveru.
- 9) Server túto správu prijíme.

- 10) Serverová aplikácia načíta vstup z klávesnice a vykoná akcie spojené s danými klávesami od klienta aj servera.
- 11) Server skontroluje, či prijatá správa nemá nastavenú premennú na ukončenie hry. Ak áno, skočíme na krok 13); inak hra pokračuje ďalej.
- 12) Nakoniec sa vykreslia súradnice prvkov pre oboch hráčov. Teraz sa vrátíme na krok 6).
- 13) Hra sa ukončí pre oboch hráčov (preruší sa cyklus) a vyhodnotí.
- 14) Obom hráčom sa výsledok hry vypíše na obrazovku a okno hry sa im zatvorí.
- 15) Klient aj server upracú operačnú pamäť od dynamicky alokovaných štruktúr a vypnú "SDL2" knižnicu.

Testovanie

Hru som testoval na úniky pamäte (memory leaks) nástrojom “Valgrind”. Testoval som serverovú a klientskú aplikáciu najprv po dobu približne 10 minút , a potom po dobu približne 20 minút. Chcel som tak zistiť, či je veľkosť únikov pamäte závislá od dĺžky spustenia programu. Výpisy pre serverovú a klientskú aplikáciu (v takomto poradí) z nástroja “Valgrind” sú uvedené nižšie.

Serverová aplikácia – Valgrind výpis 1

```
andrej@hp:~/Desktop/infinite_pong_final$ time valgrind --leak-check=yes ./infinite_pong_server 8888
==8922== Memcheck, a memory error detector
==8922== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==8922== Using Valgrind-3.10.0 and LibVEX; rerun with -h for copyright info
==8922== Command: ./infinite_pong_server 8888
==8922==
The game shall begin!
Setting up the game ...
Koniec hry
51:33
Vyhral si! :D
==8922==
==8922== HEAP SUMMARY:
==8922==   in use at exit: 74,423 bytes in 514 blocks
==8922== total heap usage: 1,705,705 allocs, 1,705,191 frees, 6,121,153,830 bytes allocated
==8922==
==8922== 33 bytes in 1 blocks are definitely lost in loss record 14 of 101
==8922==   at 0x4C28C20: malloc (vg_replace_malloc.c:296)
==8922==   by 0x636759C: XGetWindowProperty (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922==   by 0x4EE7E01: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922==   by 0x4EE85CF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922==   by 0x4E7C456: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922==   by 0x4E7C4B3: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922==   by 0x4018E0: processEvents (infinite_pong_server.c:118)
==8922==   by 0x402A99: main (infinite_pong_server.c:628)
==8922==
==8922== 42 bytes in 2 blocks are definitely lost in loss record 29 of 101
==8922==   at 0x4C28C20: malloc (vg_replace_malloc.c:296)
==8922==   by 0x53C1989: strdup (strdup.c:42)
==8922==   by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922==   by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922==   by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922==   by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```

==8922== by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4EE2ECF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4E4D486: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x402811: main (infinite_pong_server.c:544)
==8922==
==8922== 42 bytes in 2 blocks are definitely lost in loss record 30 of 101
==8922== at 0x4C28C20: malloc (vg_replace_malloc.c:296)
==8922== by 0x53C1989: strdup (strdup.c:42)
==8922== by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922== by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922== by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922== by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==8922== by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==8922== by 0x40283D: main (infinite_pong_server.c:547)
==8922==
==8922== LEAK SUMMARY:
==8922== definitely lost: 117 bytes in 5 blocks
==8922== indirectly lost: 0 bytes in 0 blocks
==8922== possibly lost: 0 bytes in 0 blocks
==8922== still reachable: 74,306 bytes in 509 blocks
==8922== suppressed: 0 bytes in 0 blocks
==8922== Reachable blocks (those to which a pointer was found) are not shown.
==8922== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==8922==
==8922== For counts of detected and suppressed errors, rerun with: -v
==8922== ERROR SUMMARY: 3 errors from 3 contexts (suppressed: 4 from 1)

```

```

real    10m46.787s
user    3m5.044s
sys     0m10.408s

```

Serverová aplikácia – Valgrind výpis 2

```
andrej@hp:~/Desktop/infinite_pong_final$ time valgrind --leak-check=yes ./infinite_pong_server 8888
```

```
==6677== Memcheck, a memory error detector
```

```
==6677== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
```

```
==6677== Using Valgrind-3.10.0 and LibVEX; rerun with -h for copyright info
```

```
==6677== Command: ./infinite_pong_server 8888
```

```
==6677==
```

```
The game shall begin!
```

```
Setting up the game ...
```

```
Koniec hry
```

```
245:246
```

```
Prehral si :(
```

```
==6677==
```

```
==6677== HEAP SUMMARY:
```

```
==6677==    in use at exit: 74,423 bytes in 514 blocks
```

```
==6677== total heap usage: 3,583,077 allocs, 3,582,563 frees, 16,713,114,570 bytes allocated
```

```
==6677==
```

```
==6677== 33 bytes in 1 blocks are definitely lost in loss record 14 of 101
```

```
==6677==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==6677==    by 0x636759C: XGetWindowProperty (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6677==    by 0x4EE7E01: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4EE85CF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4E7C456: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4E7C4B3: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4018E0: processEvents (infinite_pong_server.c:118)
```

```
==6677==    by 0x402A99: main (infinite_pong_server.c:628)
```

```
==6677==
```

```
==6677== 42 bytes in 2 blocks are definitely lost in loss record 29 of 101
```

```
==6677==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==6677==    by 0x53C1989: strdup (strdup.c:42)
```

```
==6677==    by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6677==    by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6677==    by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6677==    by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6677==    by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4EE2ECF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x4E4D486: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6677==    by 0x402811: main (infinite_pong_server.c:544)
```



```
==6677==
==6677== 42 bytes in 2 blocks are definitely lost in loss record 30 of 101
==6677==   at 0x4C28C20: malloc (vg_replace_malloc.c:296)
==6677==   by 0x53C1989: strdup (strdup.c:42)
==6677==   by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==6677==   by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==6677==   by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==6677==   by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
==6677==   by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==6677==   by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==6677==   by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
==6677==   by 0x40283D: main (infinite_pong_server.c:547)
==6677==
==6677== LEAK SUMMARY:
==6677==   definitely lost: 117 bytes in 5 blocks
==6677==   indirectly lost: 0 bytes in 0 blocks
==6677==   possibly lost: 0 bytes in 0 blocks
==6677==   still reachable: 74,306 bytes in 509 blocks
==6677==     suppressed: 0 bytes in 0 blocks
==6677== Reachable blocks (those to which a pointer was found) are not shown.
==6677== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==6677==
==6677== For counts of detected and suppressed errors, rerun with: -v
==6677== ERROR SUMMARY: 3 errors from 3 contexts (suppressed: 4 from 1)
```

```
real    23m12.868s
user    7m7.244s
sys     0m21.128s
```

Klientská aplikácia – Valgrind výpis 1

```
andrej@hp:~/Desktop/infinite_pong_final$ time valgrind --leak-check=yes ./infinite_pong_client  
localhost 8888
```

```
==8923== Memcheck, a memory error detector
```

```
==8923== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
```

```
==8923== Using Valgrind-3.10.0 and LibVEX; rerun with -h for copyright info
```

```
==8923== Command: ./infinite_pong_client localhost 8888
```

```
==8923==
```

```
Koniec hry - server
```

```
51:33
```

```
Prehral si :(
```

```
==8923==
```

```
==8923== HEAP SUMMARY:
```

```
==8923==    in use at exit: 74,390 bytes in 513 blocks
```

```
==8923== total heap usage: 1,705,720 allocs, 1,705,207 frees, 6,121,130,187 bytes allocated
```

```
==8923==
```

```
==8923== 42 bytes in 2 blocks are definitely lost in loss record 28 of 100
```

```
==8923==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==8923==    by 0x53C1989: strdup (strdup.c:42)
```

```
==8923==    by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4EE2ECF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4E4D486: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x401BFC: main (infinite_pong_client.c:269)
```

```
==8923==
```

```
==8923== 42 bytes in 2 blocks are definitely lost in loss record 29 of 100
```

```
==8923==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==8923==    by 0x53C1989: strdup (strdup.c:42)
```

```
==8923==    by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==8923==    by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923==    by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==8923== by 0x401C28: main (infinite_pong_client.c:272)
==8923==
==8923== LEAK SUMMARY:
==8923== definitely lost: 84 bytes in 4 blocks
==8923== indirectly lost: 0 bytes in 0 blocks
==8923== possibly lost: 0 bytes in 0 blocks
==8923== still reachable: 74,306 bytes in 509 blocks
==8923== suppressed: 0 bytes in 0 blocks
==8923== Reachable blocks (those to which a pointer was found) are not shown.
==8923== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==8923==
==8923== For counts of detected and suppressed errors, rerun with: -v
==8923== ERROR SUMMARY: 2 errors from 2 contexts (suppressed: 4 from 1)
```

```
real    10m45.643s
user    3m11.432s
sys     0m8.088s
```

Klientská aplikácia – Valgrind výpis 2

```
andrej@hp:~/Desktop/infinite_pong_final$ time valgrind --leak-check=yes ./infinite_pong_client  
localhost 8888
```

```
==6681== Memcheck, a memory error detector
```

```
==6681== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
```

```
==6681== Using Valgrind-3.10.0 and LibVEX; rerun with -h for copyright info
```

```
==6681== Command: ./infinite_pong_client localhost 8888
```

```
==6681==
```

```
Koniec hry - server
```

```
245:246
```

```
Vyhral si! :D
```

```
==6681==
```

```
==6681== HEAP SUMMARY:
```

```
==6681==    in use at exit: 74,390 bytes in 513 blocks
```

```
==6681== total heap usage: 3,582,913 allocs, 3,582,400 frees, 16,713,052,643 bytes allocated
```

```
==6681==
```

```
==6681== 42 bytes in 2 blocks are definitely lost in loss record 28 of 100
```

```
==6681==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==6681==    by 0x53C1989: strdup (strdup.c:42)
```

```
==6681==    by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4EE2ECF: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4E4D486: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x401BFC: main (infinite_pong_client.c:269)
```

```
==6681==
```

```
==6681== 42 bytes in 2 blocks are definitely lost in loss record 29 of 100
```

```
==6681==    at 0x4C28C20: malloc (vg_replace_malloc.c:296)
```

```
==6681==    by 0x53C1989: strdup (strdup.c:42)
```

```
==6681==    by 0x63AC6C6: ??? (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x63AD4F4: _XimSetICValueData (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x63A8A28: _XimLocalCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x638DE54: XCreateIC (in /usr/lib/x86_64-linux-gnu/libX11.so.6.3.0)
```

```
==6681==    by 0x4EEFB30: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4EEFF45: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681==    by 0x4EE30CB: ??? (in /usr/lib/x86_64-linux-gnu/libSDL2-2.0.so.0.2.0)
```

```
==6681== by 0x401C28: main (infinite_pong_client.c:272)
==6681==
==6681== LEAK SUMMARY:
==6681== definitely lost: 84 bytes in 4 blocks
==6681== indirectly lost: 0 bytes in 0 blocks
==6681== possibly lost: 0 bytes in 0 blocks
==6681== still reachable: 74,306 bytes in 509 blocks
==6681== suppressed: 0 bytes in 0 blocks
==6681== Reachable blocks (those to which a pointer was found) are not shown.
==6681== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==6681==
==6681== For counts of detected and suppressed errors, rerun with: -v
==6681== ERROR SUMMARY: 2 errors from 2 contexts (suppressed: 4 from 1)

real    23m11.463s
user    7m17.720s
sys     0m16.424s
```

Vyhodnotenie testovania

Z testovania klientskej aj serverovej aplikácie vyplýva, že v oboch aplikáciách úniky pamäte existujú. Tie nie sú spôsobené nevhodnou správou pamäte z mojej strany, ale použitou knižnicou “SDL2” a knižnicou zobrazovacieho servera “Xorg” - “Xlib”, s ktorým “SDL2” knižnica spolupracuje.

Snažil som sa použiť aj univerzálne dátové typy “int32_t” z knižnice “stdint.h”, avšak to vytváralo ešte viac únikov pamäte pri použití knižnice “SDL2”. Nakoniec som sa rozhodol pre jednoduchosť a kompatibilitu s knižnicou “SDL2” použiť štandardný celočíselný typ “int”, ktorý vytváral najmenej únikov pamäte. Úniky pamäte boli v takomto prípade malé (rádovo niekoľko desiatok bajtov).

Pri hľadaní, prečo sa úniky pamäte dejú, som narazil na niekoľko linkov, ktoré hovoria, že sa týmto únikom nedá zabrániť, aj keď korektne vypneme SDL v programe, viď:

<http://stackoverflow.com/questions/24219887/big-memory-leak-in-sdl-init>

<http://stackoverflow.com/questions/1997171/why-does-valgrind-say-basic-sdl-program-is-leaking-memory>

<http://sdl.libsdl.narkive.com/mMaqsecM/why-sdl-is-reported-memory-leak-by-valgrind>

Zdroje

https://www.libsdl.org/projects/SDL_ttf/

<https://packages.debian.org/search?searchon=sourcenames&keywords=freetype>

http://lazyfoo.net/tutorials/SDL/06_extension_libraries_and_loading_other_image_formats/linux/index.php

http://lazyfoo.net/tutorials/SDL/06_extension_libraries_and_loading_other_image_formats/linux/codeblocks/index.php

<http://www.dafont.com/digital-7.font>

<https://wiki.libsdl.org/>

<https://www.youtube.com/watch?>

[v=JPAyj85tJ5E&list=PLT6WFYYZE6uLMcPGS3qfpYm7T_gViYMMt](https://www.youtube.com/watch?v=JPAyj85tJ5E&list=PLT6WFYYZE6uLMcPGS3qfpYm7T_gViYMMt)

(posledný link mi pomohol asi najviac – všetky videá z playlistu)