


Ćwiczenie 14

Pracujemy w katalogu `JAVA/14`, będzie potrzebny plik `Makefile` z poprzednich zajęć.

Ćwiczenie polega na przygotowaniu prostego tekstowego komunikatora internetowego składającego się z dwóch programów: (1) "serwera" oczekującego na połączenia nadchodzące z sieci; (2) "klienta" łączącego się z "serwerem". Na początek proszę przetestować programy uruchamiając je oba w osobnych konsolach na komputerze, przy którym pracujemy i łącząc się z klienta na adres localhost-a (127.0.0.1).

Założenia/wymagania:

1. [3 p.](#) Po uruchomieniu serwer oczekuje na połączenie na porcie `4321`; po pojawieniu się nadchodzącego połączenia ma wypisać komunikat `[connection from IP]` (jako IP podając oczywiście prawdziwy adres IP klienta); z uruchomionego serwera nie można zainicjować połączenia, ale można odpowiadać na nadchodzące od jakiegoś klienta wiadomości wypisywane w formacie `[nick] treść wiadomości`; po zakończeniu rozmowy serwer ma przechodzić w stan oczekiwania na następne połączenie.
2. [3 p.](#) Po uruchomieniu klient pyta o adres IP serwera, z którym chcemy się połączyć; następnie wysyła do serwera wiadomość o treści:
`HELLO_FROM_CLIENT nick`
gdzie `nick` to nazwa pod jaką będzie nas widział serwer; jeśli serwer odpowie `HELLO_FROM_SERVER nick-serwera` to przechodzimy do wymiany wiadomości - na początek może to działać tak, że każda strona wysyła naprzemiennie po jednej wiadomości, a wpisanie `EXIT` przez którąkolwiek ze stron zamyka połączenie; również jeśli pierwsza odebrana (zarówno przez serwer jak i klienta) wiadomość nie jest w formacie `HELLO...` to zamknąć połączenie wysyłając wcześniej wiadomość o treści `MESSAGE_ERROR`. (Można wypisywać na ekran te specjalne wiadomości odpowiadające na nawiązanie i zakończenie komunikacji.)
3. [2 p.](#) Jeżeli zarówno z serwera jak i z klienta można wysyłać kolejne wiadomości nie czekając na odpowiedź, oraz są one niezwłocznie wyświetlane u rozmówcy.
4. [2 p.](#) Jeżeli serwer jest w stanie obsługiwać jednocześnie rozmowę z kilkoma klientami; odpowiedzi do nich rozróżnimy poprzedzając je napisem `@nick`.



```

$ java Server
[connection from 127.0.0.1]
[alfa] HELLO_FROM_CLIENT alfa
HELLO_FROM_SERVER beta

[alfa] abc def
ghi jkl
[alfa] xyz
EXIT

$ java Client
IP = 127.0.0.1
HELLO_FROM_CLIENT a
[beta] HELLO_FROM_S

abc def
[beta] ghi jkl
xvz
[beta] EXIT

$

```

Przed wyjściem z sali proszę przesłać plik(i) z wykonanym ćwiczeniem uruchamiając polecenie `make send`.

About this page...

Last modified: Mon, 17 Jan 2022 22:56:56 GMT.

woloszyn@newton.fis.agh.edu.pl

