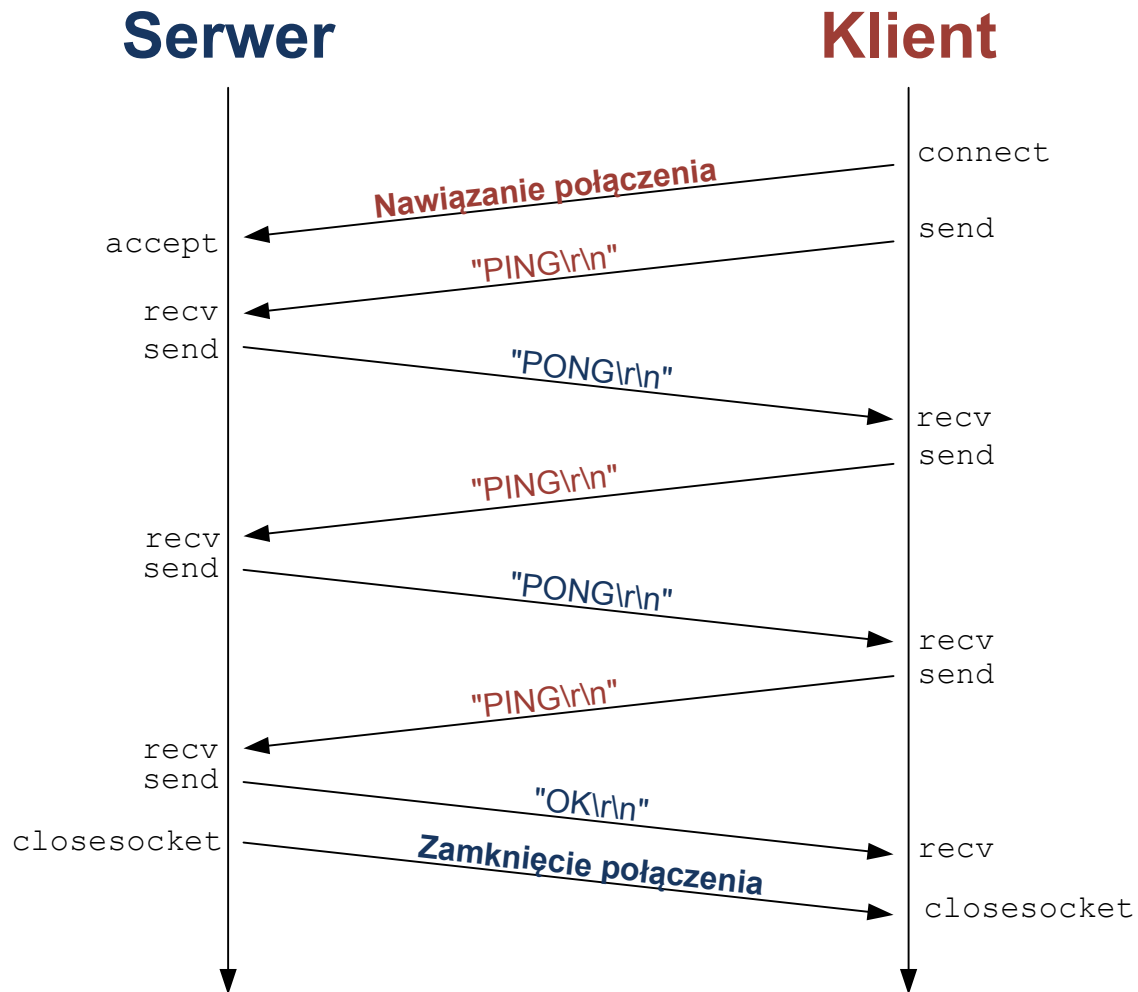


## Zaliczenie przedmiotu Programowanie sieciowe 1 LABORATORIUM ORAZ WYKŁAD

### Zaliczenie przedmiotu *Programowanie Sieciowe 1* na ocenę 3.0:

Aby zaliczyć przedmiot na ocenę 3.0, należy wykonać parę serwer -klient PING realizującą komunikację według poniższej specyfikacji:



Rysunek 1 Komunikacja klient-serwer PING-PONG

Serwer może być aplikacją wielowątkową. Może pracować w konsoli lub jako okno dialogowe. Język programowania wykorzystany do jego stworzenia – dowolny. Do testowania serwera można wykorzystać aplikację *telnet*.

#### Opis działania serwera:

Serwer ma skonfigurować system gniazdowy (funkcje *socket*, *bind*, *listen*) oraz nasłuchiwać w pętli. Po nadejściu połączenia od klienta, serwer oczekuje na kolejne komunikaty. Serwer powinien przyjąć każdy tekst przesłany do niego, jednak odpowiedzieć

może tylko na linię zawierającą komunikat **PING**<sup>1</sup>. Dla danego klienta serwer odpowiada dwa razy na komunikat **PING** komunikatem **PONG**. Przy trzecim odebraniu komunikatu **PING** od klienta, serwer odpowiada komunikatem **OK** oraz zamyka połączenie. **Serwer nie może wysyłać żadnych danych/komunikatów bez wcześniejszego odebrania żądania od klienta.**

### Opis działania klienta:

Klient ma za zadanie połączyć się z serwerem a następnie wysłać mu kolejno 3 komunikaty **PING**. Pierwszy komunikat ma zostać wysłany zaraz po połączeniu. Dwa kolejne natomiast, jako odpowiedź na komunikat **PONG** nadesłany przez serwer<sup>2</sup>. W ten sposób powstaje swoista sytuacja „odbijania piłeczki” (PING->PONG->PING->PONG->PING->OK) pokazana na rys. 1. Wysłane i odebrane komunikaty powinny być wyświetlane na ekranie w postaci tekstów „PING”, „PONG” oraz „OK”, jak w przykładzie:

```
Polaczyłem sie z serwerem 127.0.0.1:3301...
```

```
Wysyłam: PING
```

```
Otrzymałem: PONG
```

```
Wysyłam: PING
```

```
Otrzymałem: PONG
```

```
Wysyłam: PING
```

```
Otrzymałem: OK
```

```
Polaczenie zostalo zamkniete
```

### Elementy oceny serwera oraz klienta:

- ⇒ Serwer odpowiada tylko na komunikat **PING**. W przypadku odebrania innego komunikatu, serwer może je zignorować lub natychmiast zerwać połączenie.
- ⇒ Na pierwszy oraz drugi komunikat **PING** serwer odpowiada komunikatem **PONG**. Na trzeci komunikat **PING** serwer odpowiada komunikatem **OK**.
- ⇒ Zamknięcie połączenie ma odbywać się po stronie serwera.
- ⇒ Klient reaguje na komunikaty nadsyłane przez serwer i na komunikat **PONG** odpowiada komunikatem **PING**, realizując „odbijanie piłeczki” (jak pokazano na rysunku 1)
- ⇒ Znajomość poszczególnych elementów kodu klienta oraz serwera.
- ⇒ **Student nie otrzyma zaliczenia, jeśli klient lub serwer będą wysyłały odpowiednio PING/PONG bez oczekiwania na wcześniejsze zdarzenia, jak na rysunku 1.**

Na mojej stronie umieściłem również gotowe (skompilowane) wersje serwera oraz klienta PING/PONG:

[http://tjaworski.kis.p.lodz.pl/ps1/wykonywalne/server\\_ping.exe](http://tjaworski.kis.p.lodz.pl/ps1/wykonywalne/server_ping.exe)

[http://tjaworski.kis.p.lodz.pl/ps1/wykonywalne/client\\_ping.exe](http://tjaworski.kis.p.lodz.pl/ps1/wykonywalne/client_ping.exe)

Programy te można wykorzystać do testowania napisanych przez Państwa wersji klientów i serwerów. Proszę również zwrócić uwagę na działanie programu serwerowego. Aby to zrobić, proszę połączyć się programem *telnet* na adres **127.0.0.1** oraz port **3301**. W tym celu należy wybrać z menu **Start** polecenie **Uruchom** lub wcisnąć kombinację klawiszy **Windows+R**, a następnie wpisać „telnet 127.0.0.1 3301”. W oknie programu

<sup>1</sup> Podpowiedź: Nie trzeba przysyłać tekstu „PING”, „PONG” oraz „OK”. Stworzony protokół może wykorzystywać identyfikatory zamiast komunikatów, np. PING=1, PONG=2, OK=3. Zastosowanie jednego bajtu do przesyłania takiego komunikatu może znacznie ułatwić zadanie.

<sup>2</sup> Wysłanie trzech poleceń PING bez oczekiwania na odpowiedzi PONG oznacza brak zgodności z zadaniem i brak zaliczenia laboratorium.

---

*telnet*, wpisując różne komunikaty (potwierdzone klawiszem ENTER) oraz komunikat „PING”, proszę obserwować informacje wyświetlane w oknie programu serwera.

**Zaliczenie przedmiotu *Programowanie Sieciowe 1* na ocenę 3.5 lub wyższą:**

Aby uzyskać ocenę 3.5 lub wyższą należy przygotować projekt o temacie wcześniej uzgodnionym z prowadzącym. Temat można ustalić w trakcie zajęć lub mailowo. Projekt należy oddać do **końca semestru lub w sesji (proszę pamiętać o terminie złożenia indeksów do dziekanatu!!)**. Można go również zaprezentować na zajęciach laboratoryjnych. Przykładowe tematy:

1. Czat pozwalający na skomunikowanie się dwóch osób (sugerowany protokół TCP),
2. Czat pozwalający na skomunikowanie się wielu osób (sugerowany protokół UDP),
3. Przesyłanie pliku (klient podaje nazwę pliku a serwer przesyła jego zawartość; sugerowany protokół TCP),
4. Gra internetowa dla dwóch graczy (np. statki, kółko i krzyżek, itp.; sugerowany protokół TCP),
5. Przesyłanie obrazu z kamery internetowej do zdalnego komputera (sugerowany protokół TCP),
6. Przesyłanie obrazu dźwięku z mikrofonu do zdalnego komputera (sugerowany protokół TCP),
7. Zdalny pulpit – klient wysyła informacje o położeniu myszy i stanie przycisków a serwer wysyła zrzuty ekranu (podobnie do programu VNC; sugerowany protokół TCP),

Projekt może dotyczyć również każdego innego zastosowania zaproponowanego przez studenta. Ważne jest jednak, aby tworzony program korzystał z biblioteki *WinSock*.

**Uwaga!** Zaliczenie przedmiotu składa się z dwóch części:

1. przedstawienia działającego programu prowadzącemu
2. oraz omówienie jego działania.

Przykładowe pytania, mogące paść podczas omawiania programu:

- a. Co realizuje funkcja `recv`?
- b. Co zwraca funkcja `accept`?
- c. Co wysyła serwer a co klient?