

System mnożenia dużych macierzy kwadratowych - zaliczenie projektu z Sieci Komputerowych 2

Jacek Gulij 136714, Marek Subocz 136806

18.20.2020

1 Opis protokołu komunikacyjnego

Klient i serwer w naszym projekcie łączą się za pomocą protokołu TCP. Po nawiązaniu połączenia na porcie 8080 klient wysyła serwerowi zapytanie w postaci:

- jednej liczby całkowitej ($\text{int } n$) - rozmiar jednego wymiaru macierzy kwadratowych do pomnożenia
- n^2 liczb zmiennoprzecinkowych (double) - elementy pierwszej macierzy, podane rzędami
- n^2 liczb zmiennoprzecinkowych (double) - elementy drugiej macierzy

Następnie serwer, po obliczeniu wyniku mnożenia obu macierzy, odsyła odpowiedź w postaci n^2 liczb zmiennoprzecinkowych - elementów macierzy wynikowej.

2 Opis implementacji

2.1 Serwer

Serwer ma zaimplementowane połączenie **na socketach** z pomocą bibliotek "netinet/in.h" i "sys/socket.h". Po utworzeniu socketu i zajęciu portu 8080 serwer zaczyna nasłuchiwanie. Kiedy pojawi się połączenie, akceptuje je i tworzy forka. Jedna z gałęzi przeprowadza operację mnożenia macierzy i wysłanie odpowiedzi klientowi, druga natychmiast zamyka połączenie i rozpoczyna ponowne nasłuchiwanie, żeby serwer mógł przyjmować wiele zapytań na raz. Po otrzymaniu zapytania serwer zamienia bufer na string, wkłada otrzymane liczby w macierz (zaimplementowaną jako *vector < vector < double >>*). Następnie za pomocą biblioteki "Open MP" **dzieli mnożenie tych macierzy na tyle wątków, ile rdzeni ma dany komputer** i oblicza wynik. Po odesłaniu wyniku do klienta gałąź kończy swój żywot.

2.2 Klient

Klient dzieli się na dwie części: UI w C# (SK Unity Client) oraz przesyłanie danych w C++ (SK Cpp Client Connector). W kliencie Unity można uzupełnić wielkość macierzy, ich wartości, IP i port połączenia oraz zobaczyć wynik obliczeń. Przycisk "Calculate" uruchamia proces klienta C++ podając mu na standardowym strumieniu wejścia parametry połączenia oraz wartości charakteryzujące macierze. Następnie nasłuchuje na standardowym wyjściu i po zakończeniu procesu klienta C++ klient Unity wyświetla wynik. Klient C++ czytuje ze standardowego wejścia parametry połączenia oraz wartości charakteryzujące macierze, a następnie otwiera połączenie i wysyła je na podany adres IP i port. Klient C++ czeka na zakończenie obliczeń i gdy otrzyma odpowiedź wypisuje ją na standardowe wyjście.

3 Sposób kompilacji, uruchomienia i obsługi

3.1 Serwer

Serwer jest zaimplementowany w jednym pliku "server.cpp". Należy go kompilować z takimi flagami:

```
g++-9 -Wall -fopenmp server.cpp -o server.o
```

Uruchomienie to zwykłe

```
./server.o
```

3.2 Klient Unity

Standardowe budowanie projektu Unity w wersji 2019.2.11f1 lub nowszej na platformę "PC, Mac & Linux Standalone" (działa tylko na Windows). Należy uruchomić "Matrix Multiplication Client.exe" bez zmieniania lokacji.

3.3 Klient C++ (bez UI)

Należy otworzyć projekt w CodeBlocks i dokonać skonfigurowanej wcześniej kompilacji. Skompilowany Client Connector.exe znajduje się w SK Unity Client/Assets/StreamingAssets/. Kolejne parametry uruchamiania rozdzielane dowolnymi białymi znakami to

- IP
- port
- N (rozmiar macierzy)
- NxN wartości pierwszej macierzy (wartości mogą być niecałkowite o ile zawierają ".", a nie ",")
- NxN wartości drugiej macierzy.

Na wyjściu pojawi się NxN wartości macierzy, która jest wynikiem mnożenia.