

8.04.2022

Zadanie domowe (przypomnienie):

Proszę dokończyć implementację serwera TCP sumującego liczby według protokołu z zestawu 5. i przesłać ją przez formularz zadania na MS Teams do czwartku 14 kwietnia 2022.

Instrukcje na zajęcia

1. Proszę zapoznać się z przykładami serwerów obsługujących wiele połączeń jednocześnie udostępnionymi przez Wykładowcę:
 - [przykład pokazujący różne sposoby zrównoleglania pracy serwera w C](#)
 - [sterowany zdarzeniami i obiektowo zaimplementowany serwer równolegle obsługujący wiele połączeń TCP](#)

(nieobowiązkowe) Wspomniane powyżej zadanie z serwerem TCP sumującym liczby można zaimplementować w wersji obsługującej wielu klientów naraz.

W drugiej części zajęć zapoznamy się z dwoma najpowszechniej używanymi tekstowymi formatami wymiany danych: XML oraz JSON.

- Zapoznaj się krótko ze standardami dotyczącymi obydwu formatów:
 - [JSON - RFC 8259](#)
 - [XML 1.0 - W3C Recommendation](#)

Czy te standardy określają jakie typy danych (np. wartości logiczne, liczby całkowite, liczby rzeczywiste, *NULL*) mogą zostać zawarte w tych formatach?

- Przeanalizuj strukturę poniższych danych w formacie JSON. Czy na ich podstawie potrafisz zapisać powiedzieć jakie są ogólne zasady tego formatu?

```
{
  "XML": {
    "standard release date" : 1998,
    "example libraries" : [
      {"C++" : ["rapidXML", "tinyXML2"]},
      {"Python" : ["xml.etree.ElementTree"]},
      {"Java" : ["DOM Parser", "SAX Parser"]}
    ],
    "supports bool":true,
    "% of Stack Overflow questions in 2019" : 0.79
  },
  "JSON" : {
    "standard release date" : 2017,
    "example libraries" : [
      {"C++" : ["rapidJSON"]},
      {"Python" : ["json"]},
      {"Java" : ["org.json", "Google GSON"]},
      {"Fortran" : null}
    ],
    "supports bool":false,
    "% of Stack Overflow questions in 2019" : 1.45
  }
}
```

- Znajdź w swoim ulubionym języku programowania bibliotekę pozwalającą zapisywać oraz odczytywać dane w formacie JSON. Na komputerach w SPK możesz użyć np. biblioteki [rapidjson](#) w C++ albo pakietu [json](#) w Pythonie.

- Posługując się dokumentacją wybranej biblioteki stwórz program odtwarzający powyższe dane w formacie JSON i zapisujący je do pliku.
- Używając tej samej biblioteki stwórz program odczytujący plik stworzony w poprzednim punkcie. Program ten może na przykład wypisywać wartości kilku wybranych pól, np. nazwy parserów XML z Javy.
- Zastanów się, jak wyglądałyby powyższe dane w formacie XML. Możesz spróbować ręcznie napisać odpowiedni plik XML lub od razu zacząć następny punkt i wypracować jego strukturę w trakcie tworzenia programu.
- Zrealizuj to samo ćwiczenie co w przypadku formatu JSON, ale tym razem w XML: znajdź odpowiednią bibliotekę i napisz program zapisujący powyższe dane do pliku w formacie XML.
 - Zamiast tworzyć plik od nowa, możesz też spróbować napisać program, który "przepisuje" dane z formatu JSON na XML wykorzystując swoje rozwiązanie wcześniejszego zadania.
- Następnie napisz program odczytujący stworzony plik XML i wypisujący kilka przykładowych danych.
- W oparciu o format JSON, zaprojektuj format zapytania wysyłanego przez klienta do serwera sumującego liczby oraz format odpowiedzi serwera.
- (*nieobowiązkowe*) Stwórz format zapytania i odpowiedzi serwera sumującego liczby oparty na XML. Następnie poczytaj o walidacji plików XML przy pomocy *XML Schema* (np. XSD) i spróbuj stworzyć plik *Schema* definiujący dodatkowe wymagania do zaprojektowanego formatu, np.:
 - wszystkie pola przewidziane na liczby do zsumowania mają zawierać określony typ liczbowy
 - odpowiedź serwera ma zawierać jedną liczbę z wynikiem albo informację o błędzie.