# Проект по дисциплината „Мрежово програмиране“

Имена: Кристина Желязкова

Факултетен номер: 81495

Дисциплина: Мрежово програмиране

Тема: Клиент-сървър приложение чрез UDP протокола

# Съдържание

* Тема
* Теоретично описание на решението на проблема
* Проектиране
* Реализация
* Ръководство за инсталация
* Резултати
* Литература
* Приложение

# Тема

Реализирани са клиент и сървър приложения чрез UDP протокола за търсене на зависимост на поведение на потребител.

Клиентът изпраща до сървър лог файл със структура:

Time;Event context;Component;Event name;Description;Origin;IP address

Сървърът анализира поведението на потребителите на база данните от файла с помощта на откриване на шаблон на данните чрез реализирането на Apriori алгоритъм за извличането на данни.

Търси се честотата (като брой дни) на срещане на множества от събития, случили се в един ден. Информацията се предоставя на клиента във вид удобен за визуализиране на зависимостите.

Използваният алгоритъм е:

<http://www.philippe-fournier-viger.com/spmf/Apriori.php>

За тестване на системата се използва файла logs\_BCS37\_20181103.csv

# Теоретично описание на решението на проблема

Клиентът изпраща на сървъра файлът като преработена информация, сървърът го приема и обработва.

Сървърът прави биективно съответствие между всички събития и числата от 1 до N, където N е броят на различните събития.

След това прави списъци какви събития са се случили за всеки от дните.

Преработва тази информация, така че да се подаде на алгоритъма Apriori, който дава информация за честотата на срещане на множества от събития в различните дни.

# Проектиране

Проектът се състои от 11 файла. 7 от тях са от библиотеката SMPF и се използват за реализиране на Apriori алгоритъма. Това е единствената използвана външна библиотека. Останалите 4 файла са: Client.java, Server.java, FileLine.java, Pair.java.

FileLine.java реализира клас за съхранение на ред от входния файл като обект. Има конструктор за общо ползване, с който клиента създава необходимите обекти.

Pair.java реализира клас за съхранение на двойката IP адрес и порт за уникална идентификация на клиента.

Първите 2 реализират съответно клиентската програма и сървърната. Те си комуникират чрез изпращане на пакети чрез UDP протокола.

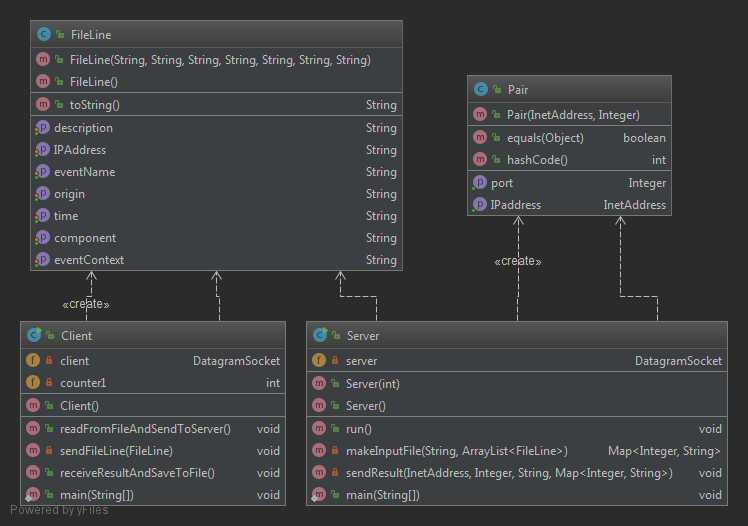
Клиентската програма чете входните данни – файла logs\_BCS37\_20181103.csv ред по ред и за всеки ред създава обект от класа FileLine, който изпраща като DatagramPacket на сървъра.

Сървъра чете получените пакети, преработва ги в обекти от клас FileLine и ги добавя като елементи в списък ArrayList, който е стойност в Map<Pair, ArrayList<FileLine>> с ключ Pair, отговарящ на клиента, от който е получен този пакет. Така за всеки клиент се пази отделна двойка в Map-а.

След като сървърът получи сигнал от клиента, че е получил цялата необходима информация, създава входния файл за алгоритъма.

Обхожда списъка от FileLine за съответния клиент и създава 3 Map-а – един със съответсвието между събитие и число, един с обратното съответствие и един със съответстието между ден и TreeSet от числа, съответстващи на събития, случили се в този ден. Чрез използването на TreeSet се избягва дублирането на събития за един ден и те се подреждат сортирано, което е изискване за входния файл на алгоритъма. След като е обходил целия списък, записва информацията по подходящ начин във файл с име input\_addr\_port.txt, където addr е IP адресът на клиента, изпратил пакетите, а port е портът, по който ги е пратил.

След това сървърът използва Apriori алгоритъма, който създава файл с търсените зависимости с име result\_addr\_port.txt.

Накрая сървърът чете получения файл от алгоритъма и използва съответствието между числа и събития, за да изпрати на потребителя информацията като текст. Всеки ред от файла се преобразува в String и се изпраща като DatagramPacket на клиента, от който е получил първоначалния файл.Клиентът получава пакетите и показва информацията в тях като текст на екрана.

# Реализация

Клиентът е реализиран чрез използването на private данна ot тип DatagramSocket, чрез която се изпращат и получават пакетите.

Има метод readFromFileAndSendToServer, който прочита входния файл, преработва го като FileLine обекти и го изпраща на сървъра. Този метод използва друг метод на класа – sendFileLine, който изпраща 1 обект от клас FileLine на сървъра.

Последният метод на този клас е receiveResultsAndPrint, който получава информацията от сървъра като DatagramPacket-и и я показва на екрана като текст.

Main методът създава клиент и изпълнява readFromFileAndSendToServer и receiveResultsAndPrint последователно за този клиент.

Сървърът наследява Thread и съответно има run метод, в който в безкраен цикъл получава пакети от клиентите.

Когато получи сигнал от някой клиент, че всички пакети от него са получени, той обработва информцията чрез метода makeInputFile и създава входния файл за Apriori алгоритъма. После прилага този алгоритъма и чрез метода sendResult праща резултата като DatagramPacket-и на клиента.

Main методът създава обект от тип Server и извиква метода start за стартирането му. Сървърът работи докато не бъде спрян външно.

Apriori алгоритъма използва входен файл, на който всеки ред се състои от цели числа, които представят транзакция, в нашия случай множество от събития. Като входна данна му се подава и цяло число, което показва колко най-малко срещания трябва да има съответното множество, за да се брои като често срещано. Като резултат алгоритъма създава файл, на който всеки ред представя едно често срещано множество, след което специален стринг и накрая броя срещания.

# Ръководство за инсталация

Създава се проект с 11-те .java файла и в src директорията се записва входния файл.

Компилират се всички файлове.

Стартира се първо Server.

После се стартира клиент Client, който показва на екрана резултата от обработката на данните в текстов вид.

Могат да се стартират още клиенти до изчерпване на свободните портове на машината.

# Резултати

В резултат на работата на приложението се получават 2 файла на машината, на която работи сървъра – входния и изходния файл от алгоритъма, а на стандартния изход при клиента се извежда информацията от изходния файл в текстов вид.

# Литература

1. SMPF – An open-source data mining library <http://www.philippe-fournier-viger.com/spmf/index.php>
2. Описание на Apriori алгоритъм - <http://www.philippe-fournier-viger.com/spmf/Apriori.php>
3. Лекциите по мрежово програмиране и по конкретно работата с UDP.
4. Примерите от упражнения за работа с UDP.
5. Stack Overflow <https://stackoverflow.com/> за търсене на решение на възникнали проблеми.
6. Java 9 for Programmers – книга за справка при използване на вградени Java класове.
7. Java 8 Pocket Guide – за бърза справка за езика Java и платформата.