УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерных технологий
Направление подготовки: Информатика и вычислительная техника

Низкоуровневое программирование

Лабораторная работа №3

Вариант 2

Выполнил:

Иевлев К. В.
Группа № P33302

Преподаватель:
Кореньков Ю. Д.

г. Санкт-Петербург

2023

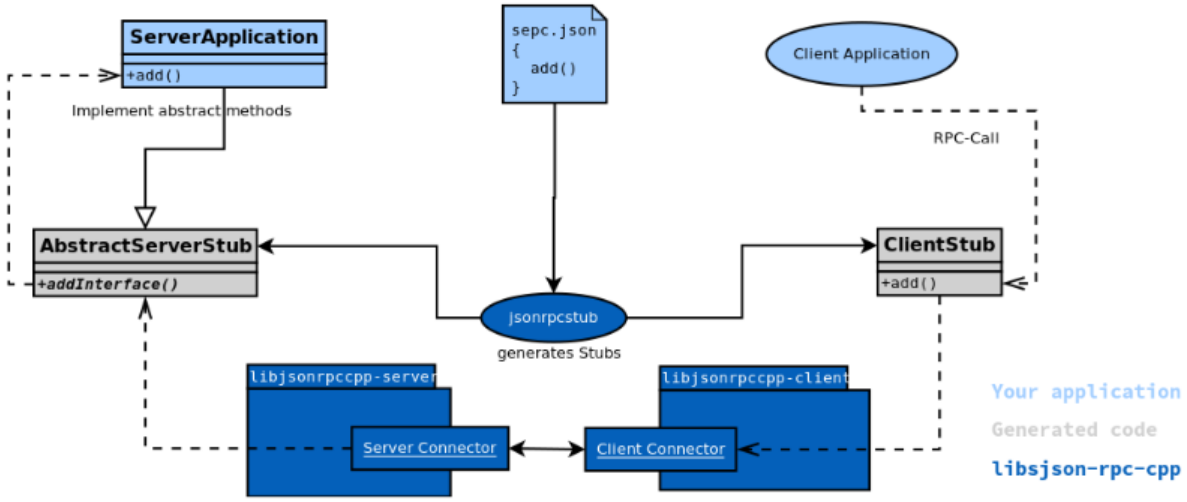
Задание:

На базе данного транспортного формата описать схему протокола обмена информацией и воспользоваться существующей библиотекой по выбору для реализации модуля, обеспечивающего его функционирование. Протокол должен включать представление информации о командах создания, выборки, модификации и удаления данных в соответствии с данной формой, и результатах их выполнения.

Используемые средства:

Библиотека json-rpc: https://github.com/cinemast/libjson-rpc-cpp
Flex
Bison

Реализация:




```
[
{
 "name" : "create__",
 "params" : {
  "tableName" : "table",
  "paramsList" : []
 },
 "returns" : "Success"
},
{
 "name" : "drop__",
```

В качестве спецификации библиотека принимает json-файл

Внутри файла массив объектов, каждый из которых отвечает за свой метод.

В описании метода есть 3 поля:

1. name - имя метода
2. params – принимаемые параметры
3. returns – возвращаемый результат

Создаются 2 файла:

1. Файл сервера, где реализуются методы
2. Файл клиента, где готовятся и собираются данные для передичи/приема

Пример функций:

```
std::string create__(const Json::Value& paramsList, const std::string& tableName) throw
(jsonrpc::JsonRpcException)
{
   Json::Value p;
   p["paramsList"] = paramsList;
   p["tableName"] = tableName;

   Json::Value result = this->CallMethod("create__",p);

   if (result.isString()) {
      std::cout << result.asString() << std::endl;
      return result.asString();
   }
   else
      throw jsonrpc::JsonRpcException(jsonrpc::Errors::ERROR_CLIENT_INVALID_RESPONSE, result.toStyledString());
}


std::string insert__(const Json::Value& paramsList, const std::string& tableName) throw
(jsonrpc::JsonRpcException)
{
   Json::Value p;
   p["paramsList"] = paramsList;
   p["tableName"] = tableName;
   Json::Value result = this->CallMethod("insert__",p);
   if (result.isString())
      return result.asString();
   else
      throw jsonrpc::JsonRpcException(jsonrpc::Errors::ERROR_CLIENT_INVALID_RESPONSE, result.toStyledString());
}

std::string MyStubServer::insert__(const Json::Value &paramsList, const std::string &tableName){
   if (!paramsList.isArray()) throw JsonRpcException(-1, "No list of arguments provided");
   if (tableName.empty()) throw JsonRpcException(-1, "Table name was empty");

   struct table* table = table_get(tableName.c_str(), this -> db);

   row* row_one = row_create(table);
   int int_val;
   double double_val;
   bool bool_val;
   string string_val;
   for (int i = 0; i <  paramsList.size() / 3; i++){
      if (paramsList[i*3].asString() == "STR") {

         string_val = paramsList[i*3 + 2].asCString();
         attribute_add(row_one,paramsList[i*3 + 1].asCString(),VARCHAR ,&string_val);

      } else if (paramsList[i*3].asString() == "INT") {

         int_val = paramsList[i*3 + 2].asInt();
         attribute_add(row_one,paramsList[i*3 + 1].asCString(),INTEGER ,&int_val);
      } else if (paramsList[i*3].asString() == "DOUBLE") {

         double_val = paramsList[i*3 + 2].asDouble();
         attribute_add(row_one,paramsList[i*3 + 1].asCString(),DOUBLE ,&double_val);
      } else if (paramsList[i*3].asString() == "BOOL") {
```

```
        bool_val = paramsList[i*3 + 2].asBool();
        attribute_add(row_one,paramsList[i*3 + 1].asCString(),BOOLEAN ,&bool_val);
      } else {

        cout << paramsList[i*2].asString()<<endl;
        return "Error!";
      }

    }
    row_insert(row_one);
    row_close(row_one);

    return std::string ("inserted into ") + std::string (tableName.c_str());
}
```

Вывод:

В процессе выполнения лабораторной работы я изучил библиотеку libjson-rpc-cpp, реализующую модель json-rps, и на ее основе был написан модуль, обеспечивающий клиен-серверное взаимодействие модулей из 1 и 2 лабораторной работы.

Задание:

База данных recommendation;

В каждой таблице нет явных ссылок на другие таблицы, поэтому я взял таблицы movie и director и отношение, которое их объединяет directed.

1 объединение таблиц director и directed

Мой вывод:

```
FOR x IN DIRECTOR FOR y IN DIRECTED FILTER name == nameDirector RETURN ALL;
```

numberOfMoviesActedIn	degree	name	movieId	year	nameDirector
9.000000	24.000000	Charles Chaplin	3309	1918	Charles Chaplin
9.000000	24.000000	Charles Chaplin	1281	1940	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3307	1931	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3629	1925	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3306	1928	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3559	1952	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3462	1936	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3310	1921	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3640	1957	Charles Chaplin
9.000000	24.000000	Charles Chaplin	8511	1917	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3632	1947	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3309	1918	Charles Chaplin
9.000000	24.000000	Charles Chaplin	1281	1940	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3307	1931	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3629	1925	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3306	1928	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3559	1952	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3462	1936	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3310	1921	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3640	1957	Charles Chaplin
9.000000	24.000000	Charles Chaplin	8511	1917	Charles Chaplin
9.000000	24.000000	Charles Chaplin	3632	1947	Charles Chaplin
4.000000	15.000000	Harold Lloyd	8423	1927	Harold Lloyd
1.000000	6.000000	Paul Wegener	25737	1920	Paul Wegener
17.000000	39.000000	Buster Keaton	25764	1928	Buster Keaton
17.000000	39.000000	Buster Keaton	83322	1921	Buster Keaton
17.000000	39.000000	Buster Keaton	83359	1921	Buster Keaton
17.000000	39.000000	Buster Keaton	83096	1921	Buster Keaton
17.000000	39.000000	Buster Keaton	8275	1927	Buster Keaton
17.000000	39.000000	Buster Keaton	25769	1928	Buster Keaton
17.000000	39.000000	Buster Keaton	83411	1922	Buster Keaton
17.000000	39.000000	Buster Keaton	7074	1924	Buster Keaton
17.000000	39.000000	Buster Keaton	8609	1923	Buster Keaton
17.000000	39.000000	Buster Keaton	83361	1922	Buster Keaton
17.000000	39.000000	Buster Keaton	3022	1926	Buster Keaton
17.000000	39.000000	Buster Keaton	3140	1923	Buster Keaton
17.000000	39.000000	Buster Keaton	25750	1924	Buster Keaton
17.000000	39.000000	Buster Keaton	83318	1921	Buster Keaton
17.000000	39.000000	Buster Keaton	3232	1925	Buster Keaton
17.000000	39.000000	Buster Keaton	3133	1925	Buster Keaton
1.000000	4.000000	Malcolm St. Clair	83318	1921	Malcolm St. Clair
4.000000	23.000000	Edward F. Cline	3140	1923	Edward F. Cline
4.000000	23.000000	Edward F. Cline	8522	1940	Edward F. Cline
4.000000	23.000000	Edward F. Cline	83411	1922	Edward F. Cline
4.000000	23.000000	Edward F. Cline	83361	1922	Edward F. Cline
4.000000	23.000000	Edward F. Cline	83359	1921	Edward F. Cline
4.000000	23.000000	Edward F. Cline	3929	1940	Edward F. Cline

```
recommendations$ MATCH p=(s:Director )-[r:DIRECTED]→(m:Movie) RETURN s.numberOfMoviesActedIn, s.degree, s.name, m.movieId, m.year
```

	s.numberOfMoviesActedIn	s.degree	s.name	m.movieId	m.year
1	9.0	24.0	"Charles Chaplin"	"3309"	1918
2	9.0	24.0	"Charles Chaplin"	"1281"	1940
3	9.0	24.0	"Charles Chaplin"	"3307"	1931
4	9.0	24.0	"Charles Chaplin"	"3629"	1925
5	9.0	24.0	"Charles Chaplin"	"3306"	1928
6	9.0	24.0	"Charles Chaplin"	"3559"	1952
7	9.0	24.0	"Charles Chaplin"	"3462"	1936
8	9.0	24.0	"Charles Chaplin"	"3310"	1921
9	9.0	24.0	"Charles Chaplin"	"3640"	1957
10	9.0	24.0	"Charles Chaplin"	"8511"	1917
11	9.0	24.0	"Charles Chaplin"	"3632"	1947
12	4.0	15.0	"Harold Lloyd"	"8423"	1927

Из Directed взял поля: name(nameDirector), year, movieId.
Из Director взял поля: name, degree, numberOfMoviesActedIn;
Почему приравнивал таблицы по name


```
{
  "start": {
    "identity": 9829,
    "labels": [
      "Actor",
      "Director",
      "Person"
    ],
```

```
    "properties": {
      "degree": 24.0,
      "name": "Charles Chaplin",
      "numberOfMoviesActedIn": 9.0
    },
    "elementId": "9829"
  },
  "end": {
    "identity": 2672,
    "labels": [
      "Movie"
    ],
    "properties": {
      "languages": [
        "English"
      ],
      "year": 1918,
      "imdbId": "0009018",
      "degree": 0.0,
      "runtime": 33,
      "imdbRating": 7.8,
      "movieId": "3309",
      "countries": [
        "USA"
      ],
      "imdbVotes": 4582,
      "title": "Dog's Life, A",
      "url": "https://themoviedb.org/movie/36208",
      "revenue": 0,
      "tmdbId": "36208",
      "plot": "The Little Tramp and his dog companion struggle to survive in the inner city.",
      "poster":
"https://image.tmdb.org/t/p/w440_and_h660_face/41vqtliesQrsJQ9iTJh5nFYQgBg.jpg",
      "released": "1918-04-14",
      "budget": 0
    },
    "elementId": "2672"
  },
  "segments": [
    {
      "start": {
        "identity": 9829,
        "labels": [
          "Actor",
          "Director",
          "Person"
        ],
        "properties": {
          "degree": 24.0,
          "name": "Charles Chaplin",
          "numberOfMoviesActedIn": 9.0
        },
```

"elementId": "9829"
},
"relationship": {
"identity": 173292,
"start": 9829,
"end": 2672,
"type": "DIRECTED",
"properties": {

},
"elementId": "173292",
"startNodeElementId": "9829",
"endNodeElementId": "2672"
},
"end": {
"identity": 2672,
"labels": [
"Movie"
],
"properties": {
"languages": [
"English"
],
"year": 1918,
"imdbId": "0009018",
"degree": 0.0,
"runtime": 33,
"imdbRating": 7.8,
"movieId": "3309",
"countries": [
"USA"
],
"imdbVotes": 4582,
"title": "Dog's Life, A",
"url": "https://themoviedb.org/movie/36208",
"revenue": 0,
"tmdbId": "36208",
"plot": "The Little Tramp and his dog companion struggle to survive in the inner city.",
"poster":
"https://image.tmdb.org/t/p/w440_and_h660_face/41vqtliesQrsJQ9iTJh5nFYQgBg.jpg",
"released": "1918-04-14",
"budget": 0
},
"elementId": "2672"
}
}
],
"length": 1.0
}

Потому что я только сейчас, писав это, понял, что “ identity ” это и есть то самое id. Но так как этот параметр имеет одинаковое название и для movie и для director, то возможно и к лучшему, что я это не заметил сразу, так как это облегчило мне парсер.

Выводы совпадают

```
kupuk@KIRILL:/home$ sudo tcpdump -i any -c 5 host 127.0.0.1
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
02:21:25.090878 lo    In  IP localhost.49704 > localhost.2323: Flags [P.], seq 390286063:390286686, ack 3983399636, win 512, options [nop,nop,TS val 3433777096 ecr 3433752819], length 623
02:21:25.090945 lo    In  IP localhost.2323 > localhost.49704: Flags [.], ack 623, win 508, options [nop,nop,TS val 3433777096 ecr 3433777096], length 0
02:21:25.096738 lo    In  IP localhost.2323 > localhost.49704: Flags [P.], seq 1:1014, ack 623, win 512, options [nop,nop,TS val 3433777102 ecr 3433777096], length 1013
02:21:25.096778 lo    In  IP localhost.49704 > localhost.2323: Flags [.], ack 1014, win 505, options [nop,nop,TS val 3433777102 ecr 3433777102], length 0
02:21:25.102808 lo    In  IP localhost.49704 > localhost.2323: Flags [P.], seq 623:806, ack 1014, win 512, options [nop,nop,TS val 3433777108 ecr 3433777102], length 183
5 packets captured
28 packets received by filter
0 packets dropped by kernel
```

С фильтрами:

```
recommendations$ MATCH p=(s:Director)-[r:DIRECTED]→(m:Movie) WHERE s.numberOfMoviesActedIn > 3 and s.degree < 24 RETURN s.numberOfMoviesActedIn, s.degree,
                 s.name, m.movieId, m.year
```

	s.numberOfMoviesActedIn	s.degree	s.name	m.movieId	m.year
1	4.0	15.0	"Harold Lloyd"	"8423"	1927
2	4.0	23.0	"Edward F. Cline"	"3140"	1923
3	4.0	23.0	"Edward F. Cline"	"8522"	1940
4	4.0	23.0	"Edward F. Cline"	"83411"	1922
5	4.0	23.0	"Edward F. Cline"	"83361"	1922
6	4.0	23.0	"Edward F. Cline"	"83359"	1921
7	4.0	23.0	"Edward F. Cline"	"3929"	1940
8	4.0	23.0	"Edward F. Cline"	"83096"	1921
9	4.0	23.0	"Edward F. Cline"	"83322"	1921
10	4.0	23.0	"Edward F. Cline"	"25874"	1941
11	9.0	20.0	"Donald Crisp"	"7074"	1924
12	4.0	15.0	"Walt Disney"	"2102"	1928

Started streaming 330 records in less than 1 ms and completed after 13 ms.

```
> FOR x IN DIRECTOR FOR y IN DIRECTED FILTER name == nameDirector && numberOfMoviesActedIn > 3.0 && degree < 24.0 RETURN ALL;
numberOfMoviesActedIn    degree                   name                     movieId                  year                     nameDirector
4.000000                 15.000000                Harold Lloyd             8423                     1927                     Harold Lloyd
4.000000                 23.000000                Edward F. Cline          3140                     1923                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          8522                     1940                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          83411                    1922                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          83361                    1922                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          83359                    1921                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          3929                     1940                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          83096                    1921                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          83322                    1921                     Edward F. Cline
4.000000                 23.000000                Edward F. Cline          25874                    1941                     Edward F. Cline
9.000000                 20.000000                Donald Crisp             7074                     1924                     Donald Crisp
4.000000                 15.000000                Walt Disney              2102                     1928                     Walt Disney
9.000000                 20.000000                Charles Laughton         7013                     1955                     Charles Laughton
8.000000                 21.000000                Ida Lupino               47092                    1951                     Ida Lupino
8.000000                 21.000000                Ida Lupino               31109                    1953                     Ida Lupino
8.000000                 21.000000                Ida Lupino               3121                     1953                     Ida Lupino
13.000000                16.000000                Laurence Olivier         7697                     1957                     Laurence Olivier
13.000000                16.000000                Laurence Olivier         25901                    1944                     Laurence Olivier
13.000000                16.000000                Laurence Olivier         1941                     1948                     Laurence Olivier
10.000000                18.000000                Orson Welles             1248                     1958                     Orson Welles
10.000000                18.000000                Orson Welles             6460                     1962                     Orson Welles
10.000000                18.000000                Orson Welles             923                      1941                     Orson Welles
10.000000                18.000000                Orson Welles             25927                    1946                     Orson Welles
10.000000                18.000000                Orson Welles             25940                    1947                     Orson Welles
10.000000                18.000000                Orson Welles             5177                     1942                     Orson Welles
10.000000                18.000000                Orson Welles             2848                     1952                     Orson Welles
13.000000                19.000000                Gene Kelly               899                      1952                     Gene Kelly
13.000000                19.000000                Gene Kelly               3606                     1949                     Gene Kelly
13.000000                19.000000                Gene Kelly               6732                     1969                     Gene Kelly
13.000000                19.000000                Gene Kelly               8999                     1976                     Gene Kelly
7.000000                 18.000000                John Ireland             69945                    1955                     John Ireland
11.000000                13.000000                Robert Mitchum           7013                     1955                     Robert Mitchum
8.000000                 20.000000                Jerry Lewis              2136                     1963                     Jerry Lewis
8.000000                 20.000000                Jerry Lewis              8986                     1960                     Jerry Lewis
22.000000                23.000000                Marlon Brando            5063                     1961                     Marlon Brando
14.000000                18.000000                Sidney Poitier           5250                     1980                     Sidney Poitier
14.000000                18.000000                Sidney Poitier           26686                    1990                     Sidney Poitier
15.000000                22.000000                Peter Sellers            2286                     1980                     Peter Sellers
8.000000                 19.000000                Anthony Perkins          2903                     1986                     Anthony Perkins
14.000000                19.000000                Steve McQueen            90531                    2011                     Steve McQueen
```

```
kupuk@KIRILL:/home$ sudo tcpdump -i any -c 5 host 127.0.0.1
[sudo] password for kupuk:
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
02:45:52.617199 lo    In  IP localhost.49712 > localhost.2323: Flags [P.], seq 887873861:887874934, ack 1030345385, win 512, options [nop,nop,TS val 3435244623 ecr 3435218410], length 1073
02:45:52.617279 lo    In  IP localhost.2323 > localhost.49712: Flags [.], ack 1073, win 504, options [nop,nop,TS val 3435244623 ecr 3435244623], length 0
02:45:52.626075 lo    In  IP localhost.2323 > localhost.49712: Flags [P.], seq 1:1014, ack 1073, win 512, options [nop,nop,TS val 3435244632 ecr 3435244623], length 1013
02:45:52.626163 lo    In  IP localhost.49712 > localhost.2323: Flags [.], ack 1014, win 505, options [nop,nop,TS val 3435244632 ecr 3435244632], length 0
02:45:52.632984 lo    In  IP localhost.49712 > localhost.2323: Flags [P.], seq 1073:1256, ack 1014, win 512, options [nop,nop,TS val 3435244638 ecr 3435244632], length 183
5 packets captured
16 packets received by filter
0 packets dropped by kernel
```

1 объединение таблиц movie и directed

```
recommendations$ MATCH p=(s:Director)-[r:DIRECTED]→(m:Movie) where  s.name is not null  RETURN  m.title, m.runtime, s.name, m.movieId, m.year ORDER BY toInteger(
                 m.movieId)
```

	m.title	m.runtime	s.name	m.movieId	m.year
1	"Toy Story"	81	"John Lasseter"	"1"	1995
2	"Jumanji"	104	"Joe Johnston"	"2"	1995
3	"Grumpier Old Men"	101	"Howard Deutch"	"3"	1995
4	"Waiting to Exhale"	124	"Forest Whitaker"	"4"	1995
5	"Father of the Bride Part II"	106	"Charles Shyer"	"5"	1995
6	"Heat"	170	"Michael Mann"	"6"	1995
7	"Sabrina"	127	"Sydney Pollack"	"7"	1995
8	"Tom and Huck"	97	"Peter Hewitt"	"8"	1995
9	"Sudden Death"	111	"Peter Hyams"	"9"	1995
10	"GoldenEye"	130	"Martin Campbell"	"10"	1995

```
FOR x IN MOVIE FOR y IN DIRECTED FILTER id == movieId RETURN ALL;
```

title	id	runtime	movieId	year	nameDirector
Toy Story	1	81	1	1995	John Lasseter
Jumanji	2	104	2	1995	Joe Johnston
Grumpier Old	3	101	3	1995	Howard Deutch
Waiting to E	4	124	4	1995	Forest Whitaker
Father of th	5	106	5	1995	Charles Shyer
Heat	6	170	6	1995	Michael Mann
Sabrina	7	127	7	1995	Sydney Pollack
Tom and Huck	8	97	8	1995	Peter Hewitt
Sudden Death	9	111	9	1995	Peter Hyams
GoldenEye	10	130	10	1995	Martin Campbell
American Pr	11	114	11	1995	Rob Reiner
Dracula: Dea	12	88	12	1995	Mel Brooks
Balto	13	78	13	1995	Simon Wells
Nixon	14	192	14	1995	Oliver Stone
Cutthroat Is	15	124	15	1995	Renny Harlin
Casino	16	178	16	1995	Martin Scorsese

```
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
03:05:37.776667 lo    In  IP localhost.49716 > localhost.2323: Flags [P.], seq 3861698174:3861698787, ack 1383838199, win 512, options [nop,nop,TS val 3436429782 ecr 3436409242], length 61
3
03:05:37.776740 lo    In  IP localhost.2323 > localhost.49716: Flags [.], ack 613, win 508, options [nop,nop,TS val 3436429782 ecr 3436429782], length 0
03:05:37.784415 lo    In  IP localhost.2323 > localhost.49716: Flags [P.], seq 1:993, ack 613, win 512, options [nop,nop,TS val 3436429790 ecr 3436429782], length 992
03:05:37.784475 lo    In  IP localhost.49716 > localhost.2323: Flags [.], ack 993, win 505, options [nop,nop,TS val 3436429790 ecr 3436429790], length 0
03:05:37.792704 lo    In  IP localhost.49716 > localhost.2323: Flags [P.], seq 613:796, ack 993, win 512, options [nop,nop,TS val 3436429798 ecr 3436429790], length 183
5 packets captured
12 packets received by filter
0 packets dropped by kernel
```

```
recommendations$ MATCH p=(s:Director)-[r:DIRECTED]→(m:Movie) where  s.name is not null and m.runtime < 100 and m.year = 1995 RETURN  m.title, m.runtime, s.name,
                 m.movieId, m.year ORDER BY toInteger( m.movieId)
```

	m.title	m.runtime	s.name	m.movieId	m.year
1	"Gone Fishin'"	94	"Christopher Cain"	"870"	1997
2	"Dangerous Ground"	95	"Darrell Roodt"	"981"	1997
3	"Shooter, The"	91	"Fred Olen Ray"	"1181"	1997
4	"Fierce Creatures"	93	"Fred Schepisi"	"1425"	1997
5	"Fierce Creatures"	93	" Robert Young"	"1425"	1997
6	"Beverly Hills Ninja"	88	"Dennis Dugan"	"1431"	1997
7	"Gridlock'd"	91	"Vondie Curtis-Hall"	"1447"	1997
8	"Pest, The"	84	"Paul Miller"	"1456"	1997
9	"Touch"	96	"Paul Schrader"	"1458"	1997
10	"That Darn Cat"	89	"Bob Spiers"	"1460"	1997

```
> FOR x IN MOVIE FOR y IN DIRECTED FILTER id == movieId && year == 1997 && runtime < 100 RETURN ALL;
title                id                   runtime              movieId              year                 nameDirector
Gone Fishin'         870                  94                   870                  1997                 Christopher Cain
Dangerous Gr         981                  95                   981                  1997                 Darrell Roodt
Shooter              1181                 91                   1181                 1997                 Fred Olen Ray
Fierce Creat         1425                 93                   1425                 1997                 Fred Schepisi
Fierce Creat         1425                 93                   1425                 1997                 Robert Young
Beverly Hill         1431                 88                   1431                 1997                 Dennis Dugan
Gridlock'd           1447                 91                   1447                 1997                 Vondie Curtis-Hall
Pest                 1456                 84                   1456                 1997                 Paul Miller
Touch                1458                 96                   1458                 1997                 Paul Schrader
That Darn Ca         1460                 89                   1460                 1997                 Bob Spiers
Vegas Vacati         1461                 93                   1461                 1997                 Stephen Kessler
Booty Call           1468                 79                   1468                 1997                 Jeff Pollack
City of Indu         1472                 97                   1472                 1997                 John Irvin
Best Men             1473                 90                   1473                 1997                 Tamra Davis
Liar Liar            1485                 86                   1485                 1997                 Tom Shadyac
```

```
kupuk@KIRILL:/home$ sudo tcpdump -i any -c 5 host 127.0.0.1
[sudo] password for kupuk:
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
03:21:25.948933 lo    In  IP localhost.49748 > localhost.2323: Flags [P.], seq 1670263038:1670264071, ack 4101760070, win 512, options [nop,nop,TS val 3437377954 ecr 3437363859], length 10
33
03:21:25.949000 lo    In  IP localhost.2323 > localhost.49748: Flags [.], ack 1033, win 504, options [nop,nop,TS val 3437377954 ecr 3437377954], length 0
03:21:25.956912 lo    In  IP localhost.2323 > localhost.49748: Flags [P.], seq 1:993, ack 1033, win 512, options [nop,nop,TS val 3437377962 ecr 3437377954], length 992
03:21:25.956979 lo    In  IP localhost.49748 > localhost.2323: Flags [.], ack 993, win 505, options [nop,nop,TS val 3437377962 ecr 3437377962], length 0
03:21:25.973341 lo    In  IP localhost.49748 > localhost.2323: Flags [P.], seq 1033:1216, ack 993, win 512, options [nop,nop,TS val 3437377979 ecr 3437377962], length 183
5 packets captured
12 packets received by filter
0 packets dropped by kernel
```

Выводы совпадают