

## РАЗРАБОТКА СТОХАСТИЧЕСКОГО ОСЦИЛЛЯТОРА

### Гаскарова Е.А.

*Гаскарова Елена Айдаровна – бакалавр,  
направление: информатика и вычислительная техника,  
Национальный исследовательский университет  
Высшей школы экономики, г. Москва*

**Аннотация:** целью данной работы является разработка стохастического осциллятора на языке C++, а значит, создать программу, которая будет выводить график в виде двух линий зависимости стоимости акций в процентном соотношении, с помощью стохастических формул.

Задачами является изучение основных понятий стохастики на работе финансового рынка, повышения навыка программирования в C++, такой как научиться работать с библиотеками CURL, которая позволяет брать данные с сайта, выявление способа как загружать котировки, определение эффективного метода решения для данной задачи.

Полученные результаты помогают идентифицировать отношение текущей цены закрытия к минимуму или максимуму за установленный период. Основные данные загружаются с сайта *finam*, а следовательно, этот стохастический осциллятор при построении графика получает обновленные данные об акциях и курсах валют.

**Ключевые слова:** анализ, маркетинг.

#### **Теоретическое введение**

Стохастический осциллятор – это индикатор, который применяется в техническом анализе, показывающий положение текущей цены сравнительно диапазона цен за установленный период в прошлом. Основным смыслом этого индикатора заключается в нахождении тех ситуаций, когда необходимо покинуть рынок до трансформаций тренда, либо когда нужно войти в рынок с зарождением нового тренда. Осуществляется с помощью сигналов перекупленности и перепроданности соответствующего актива.

Стохастический осциллятор представляет из себя график из двух линий: %K (быстрый стохастик) и %D (усредненный стохастик, медленный), которые рассчитываются по формуле(1):

$$\%K_t = \frac{C_t - L_n}{H_n - L_n} \times 100 \quad (1)$$

$C_t$  - цена закрытия текущего периода

$L_n$  - самая низкая цена за конечные  $n$  периодов

$H_n$  - самая высокая цена за конечные  $n$  периодов.

%D – скользящая средняя относительно быстрой стохастики с небольшим периодом усреднения.

Актуальность разработки стохастического осциллятора состоит в том, что проанализировав данные, можно интерпретировать для действий на рынке:

- Можно ожидать падение цен, когда цены создадут последовательность новых пиков, а стохастическому индикатору не получается взмыть больше своих предыдущих максимумов, то есть можно продавать.
- Нужно покупать, если линия графика осциллятора опустится ниже оговоренного уровня, а потом поднимется выше него. Продавать – наоборот, если сначала линия графика поднимется выше, а потом упадет.
- Если линия графика быстрой стохастики поднимется выше усредненной – следуют покупать, если выше соответственно – продавать.

Си++ - на данный момент самый используемый язык программирования, который позволяет реализовать сложные эффективные алгоритмы. А с помощью библиотек QT можно построить графики, которые легко можно добавить в форму QT, без дополнительных библиотек.

#### **Постановка задачи.**

ЗАДАЧА: разработать стохастический осциллятор

ДАНО: сайт *finam*, откуда нужно загрузить котировки для исследования цены

РЕЗУЛЬТАТ: график-индикатор цены, который и является осциллятором.

СВЯЗЬ:  $\%K_t = \frac{C_t - L_n}{H_n - L_n} \times 100$        $\%D = \text{Average}(\%K)$

ПРИ: мы выбираем сами какие котировки из *finam* нам интересны для изучения, например курс доллара за месяц.

#### **Метод решения задачи.**

Получение данных и внесение в форму.

Добавляем библиотеку CURL в программную среду QT для того, чтобы загружать котировки с помощью модуля quotes. В качестве источника биржевых цен я выбрала сайт инвестиционной компании «Финам». На сайте есть возможность сохранить файл с котировками на компьютер, где можно выбрать секцию рынка, период, и другие параметры.

Рис. 1. Пример получения файла-котировок на компьютер с сайта <http://www.finam.ru/analysis/export/default.asp>

С помощью wireshark, в get запросе можно понять - как менять параметры.

No. -	Time	Source	Destination	Protocol	Info
20	25.026163	192.168.1.2	195.128.78.52	HTTP	GET /GAZP_100716_100716.txt?d=d&market=1&em=16842&df=16&mf=6&yf=2010&dt=16&mt=6&yt=2010&p=7&f=GAZP_100716_100716&e=.txt&cn=GAZP&dtf=1&tmf=1&MSOR=0&sep=3&ep=16da tf=8 HT
23	25.083845	192.168.1.2	91.203.99.45	HTTP	GET /?host=195.128.78.52&hdn=wjgtNuOTWvTs5WzCsmUGHg== HTTP/1.1
25	25.265176	195.128.78.52	192.168.1.2	HTTP	HTTP/1.1 200 OK (finam/exportfile)
27	25.322234	91.203.99.45	192.168.1.2	HTTP/XML	HTTP/1.1 200 OK

Frame 20 (774 bytes on wire, 774 bytes captured)	
Ethernet II, Src: Quantaco_a8:de:e3 (00:1e:68:a8:de:e3), Dst: T&WElect_10:19:4b (00:1e:e3:10:19:4b)	
Internet Protocol, Src: 192.168.1.2 (192.168.1.2), Dst: 195.128.78.52 (195.128.78.52)	
Transmission Control Protocol, Src Port: amx-web1inx (2930), Dst Port: http (80), Seq: 1, Ack: 1, Len: 720	
Hypertext Transfer Protocol	
Request Method: GET	
Request URI: /GAZP_100716_100716.txt?d=d&market=1&em=16842&df=16&mf=6&yf=2010&dt=16&mt=6&yt=2010&p=7&f=GAZP_100716_100716&e=.txt&cn=GAZP&dtf=1&tmf=1&MSOR=0&sep=3&ep=16da tf=8 HT	

Hex	ASCII
0030 fa f0 6b 3b 00 00 47 45 54 20 2f 47 41 5a 50 5f	..k...GE T /GAZP
0040 31 30 30 37 31 36 5f 31 30 30 37 31 36 2e 74 78	100716_1 00716,tx
0050 74 5f 64 3d 64 26 6d 61 72 6b 65 74 3d 31 26 65	??d=d&ma rket=1&e
0060 6d 3d 31 36 38 34 32 26 64 66 3d 31 36 26 6d 66	m=16842& df=16&mf
0070 3d 36 26 79 66 3d 32 30 31 30 26 64 74 3d 31 36	=6&yf=20 10&dt=16
0080 26 6d 74 3d 36 26 79 74 3d 32 30 31 30 26 70 3d	&mt=6&yt =2010&p=
0090 37 26 66 3d 47 41 5a 50 5f 31 30 30 37 31 36 5f	7&f=GAZP _100716
00a0 31 30 30 37 31 36 26 65 30 2e 74 78 74 26 63 6e	100716&e =.txt&cn
00b0 3d 47 41 5a 50 26 64 74 66 3d 31 26 74 6d 66 3d	=GAZP&dtf= 1&tmf=
00c0 31 26 4d 53 4f 52 3d 30 26 73 65 70 3d 33 26 73	1&MSOR=0 &sep=3&e
00d0 65 70 32 3d 31 26 64 61 74 66 3d 35 20 48 54 54	ep2=16da tf=8 HT

Рис. 2. Скриншот из wireshark

- em – шифр компании (например, Газпром – 16842, остальные можно посмотреть в адресной строке финама)
  - df, mf, yf – день, месяц, год начала передачи данных
  - dt, mt, yt – дата до которой нас интересуют котировки
  - p – период – частота, с которой брались данные (например каждые 30 секунд или каждый день)
- Сама же функция quotes в моей программе выглядит так:

```

33     if (curl)
34     {
35         stdTempStr="http://195.128.78.52/GAZP_080201_100208.txt?d=d&market=1&em="+string(itoa(simb, temp, 10))+
36         "&df="+string(itoa(df, temp, 10))+&mf="+string(itoa(mf, temp, 10))+&yf="+string(itoa(yf, temp, 10))+
37         "&dt="+string(itoa(dt, temp, 10))+&mt="+string(itoa(mt, temp, 10))+&yt="+string(itoa(yt, temp, 10))+
38         "&p="+string(itoa(period, temp, 10))+
39         "&f=GAZP_080201_100208&e=.txt&cn=GAZP&dtf=4&tmf=4&MSOR=0&sep=1&sep2=1&datf=5&at=1";
40
41         curl_easy_setopt(curl, CURLOPT_URL, stdTempStr.c_str());
42
43         curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, writer);
44         curl_easy_setopt(curl, CURLOPT_WRITEDATA, &buffer);
45
46         result = curl_easy_perform(curl);
47
48         _Data->append(buffer);
49     }
50
51     return result;
52 }

```

Рис. 3. Скрин программного кода функции quotes

Где static QString buffer – это Callback функция, которая собирает полученные данные.

### Результат разработки

В итоге получаем программное решение, где сначала вводим код компании, дату, периодичность, линия синего цвета – это график быстрой стохастики, линия зеленого цвета – медленной, устредненной. Исходя из этих данных можно судить о том, что следует делать с ценными бумагами – продавать или покупать.

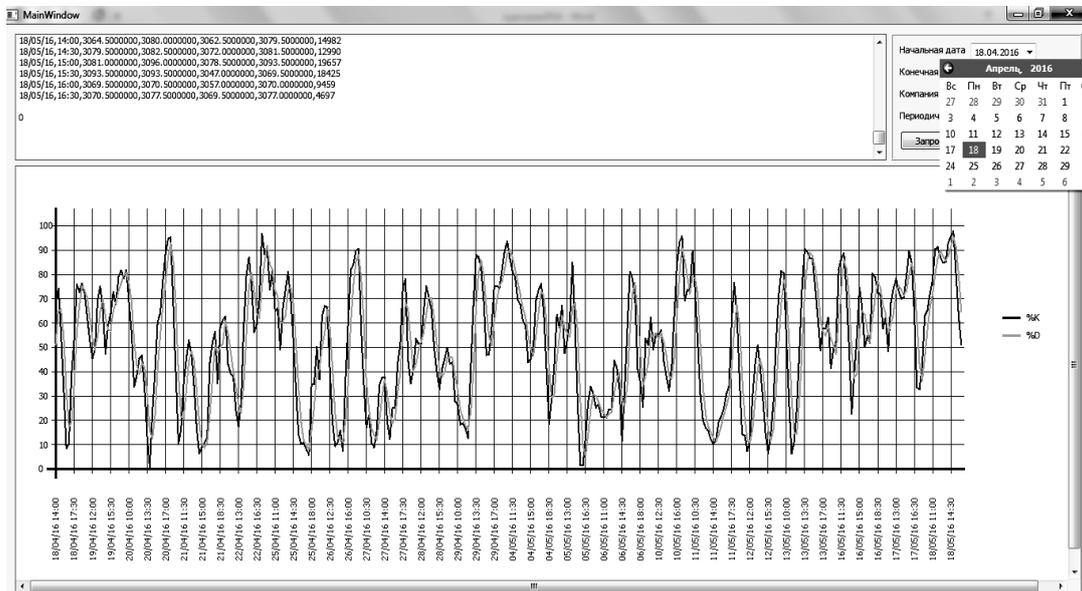


Рис. 4. Скриншот работающей программы

### Заключение.

Целью работы ставилась разработать удобную программу для пользователя, который хочет провести анализ биржевых котировок, где он сможет выбирать периодичность, компанию и частоту обновления данных. Программа работает в режиме реального времени, так как загружает самые актуальные данные с сайта finam, с технической точки зрения, осциллятор показывает абсолютно все нужные данные для анализа рынка.

«скелет» этой программы может служить для других полезных программ, которые будут обрабатывать данные из интернета.

По моему мнению эта программа очень полезной для биржевых брокеров, так как на языке C++ никто не занимался разработкой стохастического осциллятора, который можно настроить под себя и при этом безопасно анализировать данные.

### Список литературы

1. Страуструп Б. Язык программирования C++ / Б.Страуструп. Москва: Вильямс, 2011.
2. Сайт инвестиционной компании «ФИНАМ» [Электронный ресурс]. Режим доступа: <https://finam.ru> (дата обращения 13.04.2018).
3. Справочная документация по Qt [Электронный ресурс]. Режим доступа: <http://doc.crossplatform.ru/qt/4.3.2/> (дата обращения 13.04.2018).

4. Работа с библиотекой CURL. [Электронный ресурс]. Режим доступа: <http://nullflow.blogspot.ru/2011/07/c-curl.html/> (дата обращения 13.04.2018).
5. Осциллятор стохастик (Stochastic) [Электронный ресурс]. Режим доступа: <http://www.fxpro.ru/help-section/articles/fxpro-quant/stochastic-oscillator/> (дата обращения 13.04.2018).