

TRADING AUTOMATYCZNY

[...] WSPÓŁCZESNE TECHNOLOGIE WKRAČAJĄ W CORAZ TO NOWSZE OBSZARY NASZEGO CODZIENNEGO ŻYCIA [...]

WPROWADZENIE

Przyjęte oznaczenia w książce	3
Dla kogo jest ta książka?	5
Zarys historii z zakresu automatycznych strategii transakcyjnych	5

CZĘŚĆ PODSTAWOWA

Wprowadzenie do platformy Meta Trader 4.....	7
Omówienie architektury klient-serwer.....	7
Rodzaje plików w Meta Editor.....	8
Lokalizacja plików źródłowych na dysku.....	10
Podstawowe zagadnienia związane z programowaniem.....	10
Niezbędne pojęcia związane z programowaniem automatycznych strategii.....	12
Typy danych w MQL.....	12
<i>Typ int – przedstawienie liczb całkowitych</i>	13
<i>Typ bool – określenie prawdy lub fałszu</i>	13
<i>Stałe znakowe – char</i>	13
<i>Typ String – ciąg znaków</i>	14
<i>Typ double/float – liczby zmiennoprzecinkowe</i>	15
<i>Typ color</i>	15
<i>Typ Datetime– czyli jak przechować datę i czas</i>	16
<i>Typ Enum– zachowujemy porządek w kodzie</i>	17
Meta Editor jako nasze podstawowe środowisko pracy.....	18
Wprowadzenie do programowania w MQL- pierwszy Expert Advisor.....	20
Podstawy programowania.....	31
Zmienne lokalne.....	31
Zmienne globalne.....	32
Komentarze.....	32
Nazwy zmiennych oraz funkcji - identyfikatory.....	33
Funkcje.....	33
Operatory logiczne.....	42
Operatory arytmetyczne.....	44
Typy programów w Meta Editor.....	46
Co zrobić, kiedy projekt nam się rozrasta?.....	47

CZĘŚĆ ZAAWANSOWANA

Implementacja Expert Adviso.....	50
Testowanie strategii automatycznych w MQL.....	58
Funkcja automatycznej zmiany interwału czasowego.....	62

ZAKOŃCZENIE

Dodatki 67	
VPS czyli jak mieć stale uruchomiony automat mając wyłączony komputer.....	67
Najpopularniejsze błędy logiczne w automatycznych strategiach.....	68
Często używane skróty klawiszowe.....	69
Literatura.....	70



Tomasz Waszczyk

Absolwent Wydziału Automatyki, Elektroniki i Informatyki Politechniki Śląskiej w Gliwicach na kierunku Informatyka (studia prowadzone w języku angielskim).

Zawodowo zajmuję się wytwarzaniem oprogramowania oraz algorytmów samo decyzyjnych jak również wdrażaniem i utrzymywaniem zaimplementowanych rozwiązań w celu osiągnięcia jak najlepszych wyników. Do moich głównych zainteresowań należy rynek kontraktów CFD, amerykański rynek akcji oraz opcji. Codzienne obowiązki dzielę na programowanie (między innymi mobilnych aplikacji hybrydowych) oraz handel dla banku z grupy Tier 1. W czasie wolnym czytam książki o geopolityce, zarządzaniu projektami, psychologii. Jednocześnie stale zgłębiam zagadnienia współczesnej historii oraz sztuki. Uczestniczę w wielu konferencjach zarówno o tematyce oprogramowania jak i finansowej. Uważam, że nie można być długoterminowo zyskownym uczestnikiem rynku finansowego bez regularnej aktywności fizycznej - biegam, pływam oraz szukam nowych pomysłów inwestycyjnych w saunie. Dzięki systematycznemu wyszukiwaniu analogii pomiędzy światem wykresów a codzienną rzeczywistością udało mi się odkryć i wdrożyć z sukcesem parę zależności inwestycyjnych. Staram się kreować synergię pomiędzy światem wykresów a systemami informatycznymi które pozwalają na analizę dużo większej ilości danych niż mózg człowieka jest w stanie przetworzyć. Porozumiewam się w języku Mickiewicza, Szekspira oraz Schumanna.



Wprowadzenie

Każdego dnia możemy obserwować, jak współczesne technologie wkraczają w coraz to nowsze obszary naszego codziennego życia. Jednym z takich obszarów są finanse. Dawniej, aby wysłać zlecenie, należało udać się w specjalne miejsce, założyć rachunek, przelać pieniądze, wypełnić specjalny blankiet i z nim udać się do kasy naszego domu maklerskiego. Obecnie tą samą operację możemy zrobić siedząc przy własnym biurku, mając do dyspozycji jedynie komputer z dostępem do sieci Internet. Jeszcze przed paroma laty, aby móc zacząć myśleć o handlowaniu walutami na rynku międzybankowym, należało posiadać na swoim rachunku co najmniej kilkadziesiąt tysięcy waluty bazowej. Współczesne realia są zupełnie inne.

Głównym celem, który przyświecał mi przy pisaniu tej książki, było sprawienie, aby mój Czytelnik kończąc każdy podrozdział pomyślał sobie „Jakie to proste!”. Jeszcze przed rozpoczęciem pisania musiałem się zastanowić, jakiej wersji Meta Trader będzie dedykowana ta publikacja, jako że obecnie na rynku mamy dwie najpopularniejsze platformy transakcyjne Meta Trader 4 oraz nowszą jego wersję, oznaczoną jako Meta Trader 5. Ostatecznie wybrałem wersję

Meta Trader 4, uznając, że lepiej jest pisać książkę dla obowiązującego standardu. Nie odbiegniemy daleko od prawdy, stwierdzając, że nowsza wersja MT5 jest większą aktualizacją MT4. Z kolei język programowania MQL został rozwinięty o tzw. paradygmat programowania, czyli programowanie obiektowe, dzięki któremu możemy pisać kod źródłowy operując na większej abstrakcji, a jednocześnie za pomocą bardziej namacalnych obiektów.

Aby ułatwić wszystkim Czytelnikom możliwość eksperymentowania, pod adresem <https://github.com/TomaszWaszczyk/Trading-Automatyczny-MQL> znajdują się wszystkie kody źródłowe, dzięki czemu można uniknąć bezsensownego ich przepisywania do naszego Meta Editora.

W razie pytań zapraszam do kontaktu za pomocą platformy GitHub.

Przyjęte oznaczenia w książce

Aby zwiększyć czytelność treści, dla niektórych elementów, takich jak kod źródłowy, linki czy odnośniki do innych publikacji, przyjąłem następującą konwencję zapisu:

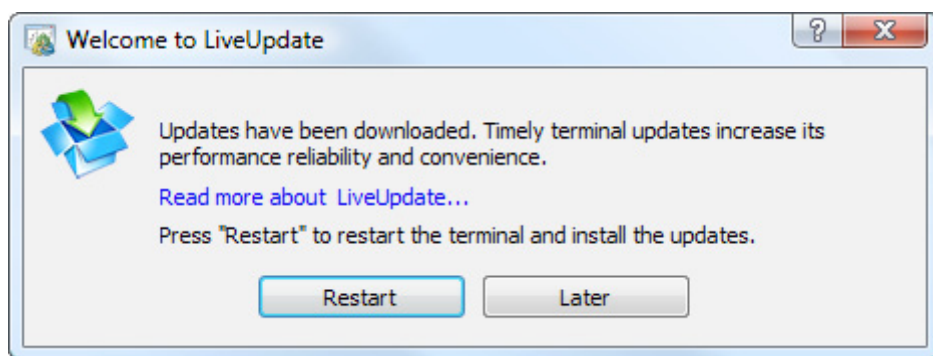
Przykładowy Kod Źródłowy:

```
int a = 50;  
int b = 100;  
MessageBox („Bardzo proste dodawanie.  $a+b=$ " + (a + b), „ $a+b=?$ ");
```

Ciekawostka: aby otworzyć Meta Editor 4, wystarczy, że naciśniesz F4 pracując w Meta Trader 4.

W związku z tym, że książka, którą, Drogi Czytelniku, trzymasz właśnie w ręku, pomyślana jest jako wstęp do programowania w MQL, chciałbym wyjaśnić, iż terminy program i system będę stosować w niej naprzemiennie. Gdy natomiast w tekście pojawi się termin strategia, przyjmuje on dwa możliwe znaczenia: po pierwsze, strategia jako ciąg wydarzeń występujących po sobie, spełniający nasze kryteria transakcyjne, czego efektem jest konkretna transakcja, po drugie strategia jako implementacja, czyli przełożenie naszych pomysłów na kod źródłowy, w celu napisania automatycznego systemu transakcyjnego. Odnosząc się do nazw Meta Trader 4 lub Meta Trader 5 stosować będę z kolei skróty MT4/MT5. Używam też tych powszechnie znanych, np. MQL – MetaQuotes Language.

Zapewne każdy z użytkowników, który na co dzień korzysta z oprogramowania Meta Trader w dowolnej jego wersji, przynajmniej raz podczas uruchomienia programu został uraczony następującym komunikatem (lub podobnym):



Jest to informacja o aktualizacji wersji MT, jaką mamy zainstalowaną u siebie na komputerze. Aktualizacje te pojawiają się stosunkowo często, co ma swoje zarówno dobre, jak i złe strony: z jednej strony powinniśmy być zadowoleni z tego, że oprogramowanie, za pomocą którego dokonujemy na co dzień decyzji inwestycyjnych, dotyczących konkretnych, w pocie czoła zarobionych przez nas pieniędzy, jest wolne od błędów. Nie chcielibyśmy przecież stracić zysków lub nawet pewnej ich części z powodu błędu w oprogramowaniu! Za pomocą aktualizacji programiści wraz z testerami wyszukują błędy, po czym je usuwają, aby każdy

z traderów mógł z zadowoleniem korzystać ze swojej platformy. Z drugiej jednak strony, czasami po aktualizacji przestają nam działać programy, jak i skrypty napisane przez nas lub innych programistów MQL. Dzieje się tak, ponieważ jedną z mniej przyjemnych konsekwencji częstych aktualizacji jest zmiana języka MQL. W efekcie, jeśli w naszym skrypcie wykorzystaliśmy funkcję, którą architekci MQL oznaczyli jako „deprecated”, czyli przestarzała (dalej już niewspierana), najczęściej oznacza to konieczność przepisania naszego kodu w taki sposób, aby był kompatybilny z najnowszą wersją naszej platformy transakcyjnej.

W związku z tymi zmianami uznałem, że potrzebujemy pewnego punktu odniesienia, czyli punktu startowego do wszelkich rozważań oraz ćwiczeń. W związku z tym będę się opierał na aktualnie najnowszej wersji MT4 (Build 745). Zainstalowaną wersję programu możemy sprawdzić klikając w głównym oknie MT w Pomoc, a następnie w zakładkę „O Programie”.

Dla kogo jest ta książka?

Zaczynając pisać tę książkę, już na samym początku musiałem odpowiedzieć sobie na bardzo ważne pytanie, kim będzie potencjalny Czytelnik. Uznałem, że Czytelnik ten jest zaznajomiony z takimi pojęciami jak cena Ask, cena Bid, spread, a także posiada podstawowe informacje na temat szeroko pojętego rynku Forex. Myślę, że krąg odbiorców można jednak poszerzyć o osoby początkujące, które znajdą w tej książce materiał do dalszego rozwoju.

Postawiłem sobie za cel, aby Czytelnik po skończeniu lektury był w stanie bez problemu napisać prosty automat i sprawnie poruszał się na platformie Meta Editor. Ma on też poznać pojęcia z dziedziny programowania MQL oraz otrzymać liczne wskazówki, zgodnie z którymi powinien dalej podążać tą drogą. Niestety książka ta ma ograniczoną objętość i nie jestem w stanie poruszyć wszystkich problemów, którym programista automatów wcześniej czy później będzie musiał stawić czoła. Wszelkie nowe pojęcia będę starał się omawiać na bieżąco, posilając się obrazkami i grafami, tak aby Czytelnik nie czuł się zagubiony już na samym początku swojej programistycznej drogi. Z doświadczenia wiem, że bardzo często pojawia się wiele pytań odnośnie programowania automatów. Chciałbym te szczególnie popularne kwestie wyjaśnić na łamach tej książki.

Myślę, że każdy zainteresowany znajdzie coś dla siebie, a być może nawet zarażę trochę osób tym fascynującym tematem. Już na wstępie chciałbym jednak jasno zasygnalizować, że budowa zyskownego automatu zabiera wiele czasu oraz energii i – mimo że często na początku udaje nam się zaprogramować automat zarabiający w testach setki procent miesięcznie – po bardziej wnikliwej analizie, nierzadko zajmującej dziesiątki godzin, znajdujemy tzw. problemy logiczne. Problemy te wydają się możliwe z technicznego punktu widzenia, jednak są niemożliwe w realnym handlu, na co warto uczulić każdego Czytelnika. Po przeczytaniu tej książki być może zaoszczędzicie ten czas, który ja straciłem, i który mogłem wykorzystać lepiej.

Zarys historii z zakresu automatycznych strategii transakcyjnych

Do napisania tego podrozdziału skłoniły mnie głównie dwie rzeczy, a jedną z nich jest z pewnością pasja do historii. Zawsze uważałem i uważam nadal, że w handlowaniu znajomość historii bardzo się przydaje, podobnie zresztą jak życiu codziennym. Drugim powodem jest często zadawane mi pytanie, czy zajmuję się programowaniem tzw. HFT Systems (High Frequency Trading), którymi są zazwyczaj komputery lub zbiór komputerów o ogromnej mocy obliczeniowej, ulokowane najczęściej w okolicach największych centrów finansowych, takich jak Londyn czy Hong Kong. Jako ciekawostkę podam przykład jednej z firm zajmującej się wytwarzaniem tego typu systemów, która na własny koszt zdecydowała się przeprowadzić łącza światłowodowe pomiędzy Londynem a Nowym Jorkiem. Inwestycja ta zwróciła się po upływie niewiele ponad roku. Odpowiadając na powyższe pytanie, zaznaczę, że nie będę opisywać tego typu programów z kilku powodów: średni budżet tego typu projektów zaczyna się od paru milionów euro i wymaga co najmniej 20-osobowego zespołu złożonego z wysoko wykwalifikowanych programistów, fizyków, statystyków oraz traderów, a i cały ten zespół niewiele zdziała, jeśli nie będzie miał dostępu do bardzo szybkich łączów pomiędzy ich komputerami (tzw. mainframe) oraz Tier 1¹. Niech zatem sam Czytelnik odpowie na pytanie: co ja mogę zrobić w pojedynkę wobec tak ogromnych funduszy inwestycyjnych, bez milionowego budżetu, zespołu ludzi czy potężnych komputerów?

Wbrew pozorom obracanie takim ogromnym kapitałem bywa bardzo problematyczne i nierzadko dużo łatwiej jest handlować mniejszymi kwotami. I tutaj widzę właśnie punkt przewagi, jaką mają traderzy „detaliczni” nad tymi „instytucjonalnymi”. Tym drugim jest bardzo trudno otworzyć właściwe pozycje, a kiedy już je otworzą, równie trudno je zamknąć, podczas gdy w przypadku automatów obracających stosunkowo małym kapitałem kapitałem problem płynności nie istnieje. Myślę, że jest to godne przemyslenia zagadnienie i skłania do działania, do czego gorąco zachęcam Cię, Drogi Czytelniku.

¹ <https://www.wikijob.co.uk/wiki/investment-banking>

Z uwagi na to, że w kodzie mamy zmienną, która określa nam interwał, zgodnie z którym skrypt będzie przełączał nam interwały czasowe, proponuję utworzyć następujące skróty:

- CTRL + 1 jako zmiana interwału M1
- CTRL + 2 jako zmiana interwału M5

Plik ze skrypcem kopiujemy z odpowiednią nazwą, np. zmienInterwałM30.

Analogicznie wykonujemy taką operację dla wszystkich interwałów, które nas interesują. Efektem końcowym jest możliwość przełączania się bardzo szybko pomiędzy interwałami jednym skrótem klawiszowym (jeden skrót klawiszowy na jeden interwał).

Zakończenie

Właśnie dotarłeś, Drogi Czytelniku, do końca mojej książki. Zapewne teraz czujesz wielki głód informacji, dlatego chciałbym Ci przekazać kilka rad i wskazówek, co zrobić dalej po skończeniu mojej lektury. Osobiście uważam, że rozwój kompetencji w przypadku programisty automatycznych systemów transakcyjnych odbywa się dwoma nurtami: „rynkowym” oraz „technologicznym”. W związku z tym, że tematem przewodnim tej publikacji było programowanie strategii, doradzę Ci kierunek technologiczny. Jak zapewne zauważyłeś podczas lektury, do tekstu dodawałem odnośniki do oficjalnej i szeroko dostępnej dokumentacji MQL, z uwagi na to, że nie byłem w stanie zamieścić wszystkich informacji na stronach tej książki. Niestety osoby nie znające języka Szekspira będą mieć trudniej, jako że zdecydowana większość dostępnej literatury oraz dokumentacja i inne materiały są dostępne w języku angielskim. Co ciekawe, istnieje również bardzo szeroka literatura przedmiotowa w języku rosyjskim.

Poruszyłem, niestety dość pobieżnie, parę problemów, takich takich chociażby opis wersjonowania naszego oprogramowania. Jeśli czujesz, że informacje są niepełne albo coś jest niezrozumiałe, proszę, wyślij mail na mój adres: tomasz@waszczyk.com. Postaram się też napisać parę artykułów jako uzupełnienie tej publikacji. Chciałbym poprosić każdego z Czytelnika, aby podzielili się ze mną wszelkimi uwagami. To dla mnie o tyle ważne, iż jest to moja pierwsza tego typu publikacja.

VPS czyli jak mieć stale uruchomiony automat mając wyłączony komputer.

Kluczową kwestią związaną z programowaniem automatycznych strategii transakcyjnych jest zapewnienie możliwości ciągłego analizowania rynku. Jedną z największych zalet EA jest możliwość analizowania rynku ciągle, w sposób nieprzerwany. Rodzi się jednak pytanie: jak to zapewnić? Niektórzy wykorzystują swoje prywatne komputery jako maszyny docelowe, na których będą handlować. Proszę sobie wyobrazić, co się stanie, kiedy zabraknie nam prądu, nagle stracimy dostęp do sieci Internetu czy nasz komputer najzwyczajniej w świecie się zawiesi. Wszystkie z wymienionych przyczyn skutkują de facto wyłączeniem naszego automatu i nie mamy nad tym jakiegokolwiek kontroli. Jeśli takich przerw w działaniu jest więcej, możemy uznać, że w ogóle nie panujemy nad naszym automatem, co jest bolesne w sytuacji, kiedy automat ma otwarte pozycje i pozostaje bez nadzoru: ani naszego osobistego, ani automatu.

Aby ustrzec się przed takimi sytuacjami, warto zainstalować Meta Tradera na serwerze VPS (Virtual Private Server). Firmy, które zajmują się tego typu usługami są przygotowane na awarie sprzętu oraz wpływ czynników od nich niezależnych, takich jak przerwy w dostępie do sieci Internet, poprzez równoległe połączenia. W momencie kiedy jedno łącze przestaje spełniać swoją rolę, następuje automatyczne przełączenie w taki sposób, że klienci nie są świadomi tego, że jedno z łączy uległo awarii. Logowanie się do serwera VPS może przebiegać za pomocą standardowego oprogramowania dostępnego w systemie MS Windows – Pulpit Zdalny (Remote Control) lub oprogramowania komercyjnego (zazwyczaj darmowego dla celów osobistych). Nie ma też problemu z zalogowaniem się na serwer przy pomocy urządzenia mobilnego. Dzięki temu rozwiązaniu jesteśmy pewni, że nawet gdy my nie mamy dostępu do Internetu, nasz automat stale monitoruje rynek. Komputer możemy spokojnie zamknąć lub wyjść z nim do kawiarni, by swobodnie porozmawiać z klientem o nowym kontrakcie.

Najpopularniejsze błędy logiczne w automatycznych strategiach

Zauważyłem, że niemalże każdy początkujący programista automatów bardzo szybko napotyka na bardzo trudne do wykrycia problemy, których przyczyny tkwią w samej logice programu. Jest to o tyle niebezpieczne, o ile błędy na poziomie kodu źródłowego zostają zauważone automatycznie. Mam tu na myśli po prostu brak pomyślnej kompilacji. Dużo większym zagrożeniem są błędy logiczne, które nie zostają wyłapane na poziomie kompilacji, a powinny zostać jak najszybciej wychwycone na poziomie testowania na rzeczywistych danych. Wielu początkujących programistów uznaje jednak, że „odkryli Graala”, dzięki czemu szybko staną się milionerami. Napiszę więc na koniec parę słów, aby mogli oni uniknąć tego typu rozczarowań na rynku inwestycyjnym.

Kiedy programujemy system transakcyjny, powinniśmy zadbać o to, aby symulowanie handlu odbywało

się w warunkach jak najbardziej zbliżonych do rzeczywistych. Co ważne, przed uruchomieniem systemu automatycznego powinniśmy wiedzieć, czego możemy się po nim spodziewać, aby uniknąć rozczarowań oraz zaskoczenia podczas rzeczywistego tradingu. Poniżej opiszę najczęściej popełniane błędy:

- Niemożliwa logika – mam na myśli sytuacje w kodzie źródłowym, które są jak najbardziej możliwe w przypadku kiedy mamy załadowane dane historyczne, a niemożliwe na rzeczywistym rynku. Przykładem takiego błędu może być podejmowanie decyzji inwestycyjnej po podstawie (n+1) świeczki, co w przełożeniu na język potoczny jest przeniesieniem się w przyszłość.
- Ignorowanie kosztów transakcyjnych – w przypadku rynku walutowego jest to zazwyczaj różnica pomiędzy Ask a Bid, czyli spread. O ile w systemach, których celem jest łapanie dłuższych ruchów cenowych, okazuje się to mało istotne, to już w przypadku handlu na tikach lub danych o niskiej granularności, może mieć fatalne skutki. Niemniej chciałbym wyrazić się jasno – brak implementacji kosztów transakcyjnych jest błędem i zawsze powinniśmy brać koszt transakcji pod uwagę, dokładnie tak samo jak w każdej innej dziedzinie biznesu.
- Przeoptymalizowanie systemu – klasyczny błąd polegający na dopasowaniu parametrów systemu transakcyjnego do danych historycznych. Problemem są tu bardzo prymitywne testery systemów, polegające np. na tym, że okres średniej (jeśli korzystamy ze średnich) jest dopasowany tak, aby otrzymać jak najwyższy zwrot z inwestycji. W rzeczywistości nie możemy zrobić takiej operacji, stąd mając system, dla którego stopa zwrotu na danych historycznych jest dużo wyższa niż w rzeczywistości, z dużym prawdopodobieństwem nie mamy szans utrzymać się na rynku.
- Niekompletne dane historyczne – może to być zaskoczeniem dla wielu Czytelników, jednak nieraz spotkałem się z sytuacją, kiedy testy zrobione na danych dostarczonych przez brokera na platformie Meta Trader bardzo różniły się od testów przeprowadzonych na własnym testerze. Problem tkwi w jakości danych historycznych. Nie polecam danych, które są standardowo dostępne i udostępniane na platformach, bo zazwyczaj są one bardzo wybrakowane, posiadają wiele luk, a im starsze, tym bardziej odbiegają od rzeczywistych cen. Dlatego proponuję wykupić za stosunkowo niewielką kwotę dane u firm, które zajmują się archiwizacją notowań.
- Zbyt mały okres testowania – należy mieć świadomość, że rynek stale ulega zmianie. Największą bolączką w programowaniu automatycznych systemów transakcyjnych jest fakt, że musimy te zmiany przewidzieć i zaprogramować system tak, aby móc dostosować się do nich. Nie należy zakładać, że system zarabiający przez ostatni kwartał będzie zarabiał już zawsze, dlatego bardzo ważne jest testowanie na możliwie najdłuższym okresie danych, aby system mógł się „wykazać” w jak najdłuższym okresie czasu.
- Niezrozumienie skali osi OY podczas czytania raportu z testera.

Odnosząc się do ostatniej kwestii opiszę następujący przypadek. Jakiś czas temu pewien klient polecił mi automat, według którego na podstawie testów uznał, że będzie zarabiał ok. 70% rocznie. Kiedy wczytałem się w kod systemu, wydawał się on bardzo rozsądny, na tyle, że postanowiłem przetestować go na własnym testerze i w oparciu o własne dane. Jaki byłem zdziwiony, kiedy system okazał się być programem najzwyczajniej tracącym pieniądze. Raz jeszcze poprosiłem o wcześniej przeprowadzone testy i „jedyną” różnicą były dwie rzeczy:

- wstępny depozyt
- wolumen poszczególnych transakcji

W przypadku gdy testy były wykonywane przy kwocie wstępnej w wysokości 1 000 000 euro system był zyskowny, były widoczne jedynie z pozoru niewielkie obsunięcia na kapitale. Kiedy zmieniłem skalę osi OY i przyjrzałem się jej, obsunięcia te wynosiły momentami 60% kapitału, jednak skala była na tyle duża, że wydawały się dużo mniejsze. Przy kwocie 10000 euro system okazał się być stratny – niestety na żywym rachunku.

Często używane skróty klawiszowe

F4 – uruchomienie MetaEditor z poziomu Meta Trader

F4 – szybkie przejście do Meta Trader z poziomu Meta Editor

F7 – kompilacja kodu źródłowego

ALT + M – lista funkcji z poziomu Meta Editor

CTRL + SPACJA – podpowiedź kontekstowa w Meta Editor

Literatura

1. Michael Lewis, Flash Boys.
2. Haim Bodek, The Problem of HFT –Collected Writings on High Frequency Trading and Stock Market, <http://haimbodek.com>
3. ALGLIB, Numerical Analysis Library dla Meta Trader 5
4. Strona internetowa: <http://docs.mql4.com>
5. Strona internetowa: <https://www.mql5.com/en/articles/81>

Copyright: Tomasz Waszczyk & Dom Maklerski TMS Brokers
S.A, Warszawa 2014.

Redakcja i korekta: Joanna Tyka

Niniejsza publikacja stanowi własność Domu Maklerskiego TMS Brokers S.A. i nie może być rozsyłana lub powielana bez pisemnej zgody. Zabrania się również dokonywania zmian w publikacji oraz czerpania jakichkolwiek korzyści z jej wykorzystania (sprzedaży, publikacji fragmentów).

Autor dołożył wszelkich starań, by zawarte w niej treści były prawdziwe i rzetelne, nie ponosi jednak odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w publikacji. Rynki OTC charakteryzują się dużym ryzykiem straty, która może przekroczyć początkowy depozyt.

Treści zawarte w niniejszej pracy nie stanowią rekomendacji w rozumieniu przepisów Rozporządzenia Ministra Finansów z dnia 19 października 2005 r., w sprawie informacji stanowiących rekomendacje dotyczące instrumentów finansowych, lub ich emitentów (Dz.U. z 2005 r. Nr 206, poz. 1715).

Partner publikacji:



