

Raport z przedmiotu Analiza Rynków Finansowych

Autor: Andrzej Kapczyński 145358

Prowadzący: dr inż. Adam Wojciechowski

Analiza kodu z serwisu MQL5.com

1. Kod źródłowy

Po dogłębnej analizie dostępnych kodów, postanowiono wybrać poniższy [program](#).

```
extern int min_gapsize = 1;
extern double lotsize_gap = 0.1;
datetime order_time = 0;

int init() { return(0); }
int deinit() { return(0); }

int start() {
    // Defining the variables to decide.
    Print("order time", order_time);
    double current_openprice = iOpen(Symbol(), PERIOD_M15, 0);
    double previous_highprice = iHigh(Symbol(), PERIOD_M15, 1);
    double previous_lowprice = iLow(Symbol(), PERIOD_M15, 1);
    double point_gap = MarketInfo(Symbol(), MODE_POINT);
    int spread_gap = MarketInfo(Symbol(), MODE_SPREAD);
    datetime current_time = iTime(Symbol(), PERIOD_M15, 0);

    // Catching the gap on sell upper gap
    if (current_openprice > previous_highprice + (min_gapsize + spread_gap) * point_gap
        && current_time != order_time) {

        int ticket = OrderSend(Symbol(), OP_SELL, lotsize_gap, Bid, 0, 0,
                                previous_highprice + spread_gap*point_gap, "", 4, 0, Red);

        order_time = iTime(Symbol(), PERIOD_M15, 0);
        Print("I am inside (sell)", order_time);

        if (ticket < 0) {
            Print("OrderSend failed with error: ", GetLastError());
        }
    }

    // Catching the gap on buy down gap
    if (current_openprice < previous_lowprice - (min_gapsize + spread_gap) * point_gap
        && current_time != order_time) {
        ticket = OrderSend(Symbol(), OP_BUY, lotsize_gap, Ask, 0, 0,
                            previous_lowprice - spread_gap*point_gap, "", 5, 0, Green);

        order_time = iTime(Symbol(), PERIOD_M15, 0);
        Print("I am inside (buy)", order_time);
        if (ticket < 0) {
            Print("OrderSend failed with error #", GetLastError());
        }
    }
    return(0);
}
```

2. Analiza kodu

Kod przedstawiony powyżej jest implementacją eksperta w języku MQL4 dla platformy MetaTrader 4. Ekspert ma za zadanie monitorować występowanie luk (gaps) na wykresach w interwale czasowym M15 oraz otwierać odpowiednie pozycje kupna lub sprzedaży na podstawie warunków zdefiniowanych w kodzie. Ekspert porównuje cenę otwarcia nowego słupka z ceną maksymalną lub minimalną poprzedniego słupka, jeśli jest luka, bezpośrednio wysyła i zamawia oraz odbiera zysk.

Kod zawiera parametry wejściowe, takie jak **min_gapsize** i **lotsize_gap**, które można dostosować, aby zgodnie z preferencjami ustawić minimalny rozmiar luki oraz wielkość pozycji handlowej. Poza tym, kod zawiera także zmienną **order_time**, która przechowuje czas ostatniego otwarcia zamówienia.

Główna funkcja **start()** jest wywoływana cyklicznie podczas działania eksperta. W jej wnętrzu znajduje się logika handlowa. Na początku funkcji, pobierane są różne informacje dotyczące cen i rynku, takie jak cena otwarcia bieżącej świecy (**current_openprice**), najwyższa cena poprzedniej świecy (**previous_highprice**), najniższa cena poprzedniej świecy (**previous_lowprice**), minimalny punkt luki (**point_gap**) oraz rozprzestnienie (**spread_gap**).

Następnie, na podstawie zdefiniowanych warunków, sprawdzane są luki na górze (sell upper gap) oraz na dole (buy down gap). Jeśli warunki są spełnione i nie wystąpiło już zamówienie w bieżącej świecy (**current_time != order_time**), odpowiednie zamówienie zostaje wysłane przy użyciu funkcji **OrderSend()**. W przypadku niepowodzenia wysłania zamówienia, wyświetlany jest komunikat z błędem w dzienniku.

Podsumowując, kod ten dostarcza prostą strategię handlową opartą na monitorowaniu i wykorzystaniu luk na wykresach. Parametry oraz logika strategii mogą zostać dostosowane według preferencji użytkownika. Warto jednak zaznaczyć, że przed zastosowaniem strategii na koncie rzeczywistym, zaleca się przeprowadzenie gruntownych testów przy pomocy testera strategii lub na koncie demonstracyjnym w celu oceny jej skuteczności i potencjalnych ryzyk.

3. Tester strategii

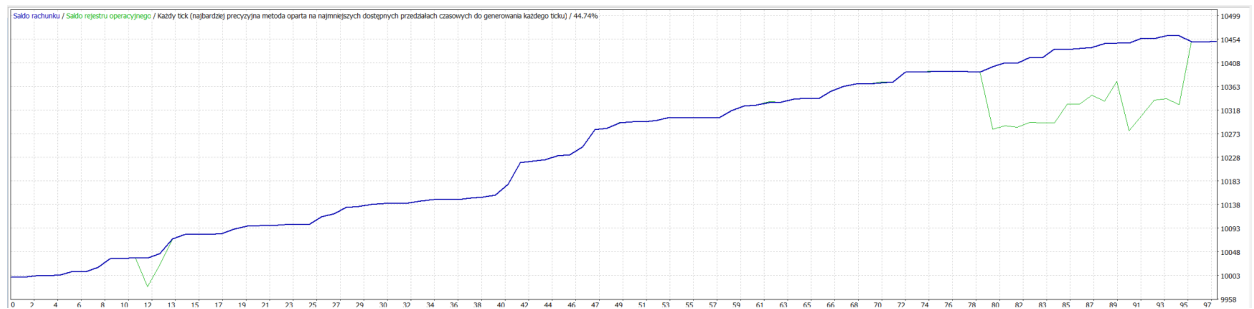
Przeprowadzono gruntowne testy przy pomocy narzędzia dostępnego w programie Meta Trader 4. Dobrano następujące ustawienia:

Strategia: ARF.ex4
Instrument: BTCUSD, Bitcoin USD
Model: Każdy tick (najbardziej precyzyjna metoda oparta na najmniejszych dostępnych przedziałach czasowych do generowania każdego ticku)
Zakres daty: Od: 2023.01.01 Do: 2023.06.05
Tryb wizualny: ☐ Przejść do: 2023.06.05
Interwał: M15
Spread: Aktualny
Optymalizacja: ☐

Uzyskano następujące wyniki:

Słupki w teście	15181	Ticki użyte w modelu	16576075	Jakość modelowania	44.74%
Błędy na wykresie	6				
Depozyt początkowy	10000.00			Spread	Aktualny (708)
Całkowity zysk netto	450.61	Zysk brutto	471.29	Strata brutto	-20.68
Wskaźnik zysku	22.79	Przewidywany zysk	4.65		
Największa strata bezwzględna	37.12	Maksymalna strata względna	207.14 (1.99%)	Strata relatywna	1.99% (207.14)
Ilość transakcji w sumie	97	Krótkie pozycje (zyskowych %)	26 (100.00%)	Długie pozycje (zyskowych %)	71 (69.01%)
		Transakcje zyskowe (% wszystkich)	75 (77.32%)	Transakcje stratne (% wszystkich)	22 (22.68%)
	Największa	zyskowa transakcja	40.78	stratna transakcja	-11.19
	Średnia	zyskowa transakcja	6.28	stratna transakcja	-0.94
	Maksymalna	ilość transakcji zyskowych pod rząd (zysk w pln)	14 (148.22)	ilość transakcji stratnych pod rząd (straty w pln)	3 (-12.55)
	Największy	nieprzerwany zysk (suma transakcji zyskowych)	148.22 (14)	nieprzerwana strata (suma transakcji stratnych)	-12.55 (3)
	Średnia	ilość transakcji zyskowych pod rząd	4	ilość transakcji stratnych pod rząd	1

W analizie wyników pomocny jest także wykres, przedstawiający saldo rachunku w stosunku do upłynionego czasu:



Na podstawie przedstawionych wyników testowania strategii przez narzędzie testowania strategii dla interwału M15 na instrumencie BTC/USD przez ostatnie 6 miesięcy, można wyciągnąć kilka wniosków:

1. Całkowity zysk netto wyniósł \$450.61, co oznacza, że strategia osiągnęła dodatni wynik finansowy w badanym okresie.
2. Jakość modelowania wynosiła 44.74%. Oznacza to, że strategia uzyskała wynik w tej samej skali, co rynek, ale nie przejawiała szczególnie wysokiej skuteczności.
3. W czasie testowania wystąpiło 6 błędów na wykresie, co sugeruje, że strategia może nie być w pełni optymalna i wymaga dalszej optymalizacji.
4. Największa strata bezwzględna wyniosła \$37.12, a maksymalna strata względna wyniosła 207.14 (1.99%). To sugeruje, że strategia może generować pewne straty, ale w granicach akceptowalnych ryzyka.
5. W analizowanym okresie zostało wykonanych łącznie 97 transakcji, z czego 75 było zyskownych (77.32%) i 22 stratne (22.68%). Większość zyskownych transakcji była długimi pozycjami (71 zyskownych transakcji, 69.01%), podczas gdy wszystkie krótkie pozycje były zyskowne (26 zyskownych transakcji, 100.00%).
6. Średnia zyskowna transakcja wyniosła \$6.28, a stratna transakcja -\$0.94. To sugeruje, że przeciętnie zyskowne transakcje są większe niż stratne transakcje.
7. Największa liczba transakcji zyskownych pod rząd wyniosła 14, generując zysk w wysokości \$148.22, podczas gdy największa liczba transakcji stratnych pod rząd wyniosła 3, generując stratę w wysokości -\$12.55. To sugeruje, że strategia miała tendencję do generowania ciągów zyskownych transakcji.

Na podstawie tych wyników można stwierdzić, że strategia oparta na kodzie powyżej osiągnęła pewien poziom sukcesu, generując dodatni wynik finansowy. Jednakże, wyniki wskazują również na możliwość poprawy strategii poprzez dalszą optymalizację i analizę błędów na wykresie. Ważne jest również zrozumienie, że przyszłe wyniki strategii mogą się różnić od wyników historycznych, dlatego zaleca się monitorowanie i dostosowywanie strategii w czasie rzeczywistym. Przed podjęciem decyzji o wykorzystaniu strategii na koncie rzeczywistym zaleca się przeprowadzenie dokładniejszej analizy ryzyka i oceny strategii na kontach demonstracyjnych.

Optymalizacja kodu

Potencjalne zmiany

Na podstawie analizy wyników testowania strategii, przygotowano kilka sugestii dotyczących potencjalnych zmian, które można rozważyć w celu poprawy wyników strategii:

- **Testowanie na większej próbie danych:** Aby uzyskać bardziej wiarygodne wyniki, zaleca się przeprowadzenie testów strategii na większej próbie danych historycznych. Dłuższy okres testowy pozwoli na lepsze zrozumienie i ocenę skuteczności strategii w różnych warunkach rynkowych.
- **Zmiana interwału czasowego:** Sugeruje się przetestowanie strategii na różnych interwałach czasowych, takich jak M30 lub H1. Badanie różnych interwałów czasowych może dostarczyć bardziej kompleksowych informacji o rynku i umożliwić lepsze dopasowanie strategii do różnych warunków rynkowych.
- **Optymalizacja parametrów:** Zaleca się przeprowadzenie optymalizacji parametrów strategii, takich jak *min_gapsize* i *lotsize_gap*, aby znaleźć optymalne wartości dla konkretnego instrumentu finansowego. Poprzez dostosowanie tych parametrów do charakterystyki rynku, można osiągnąć lepszą skuteczność strategii.

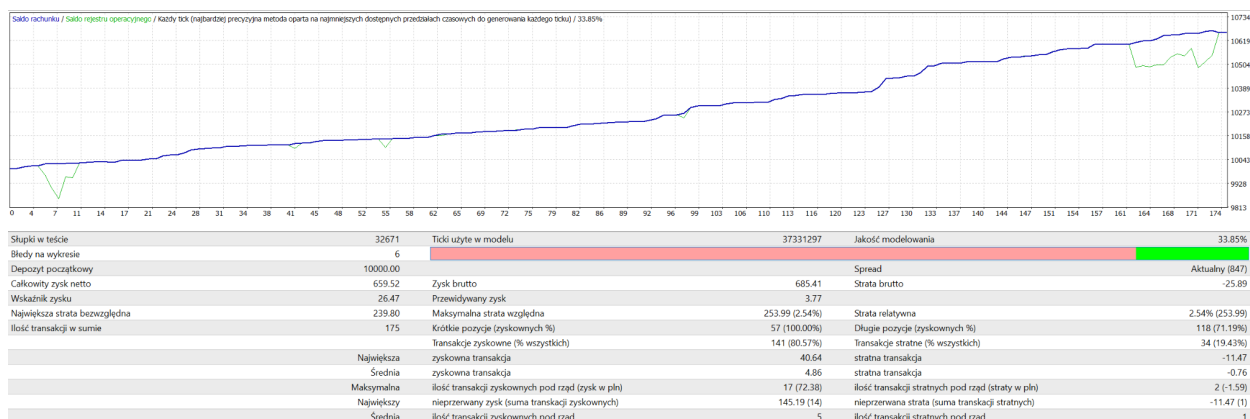
Testowanie na większej próbie danych

Postanowiono przetestować strategię na większej próbie danych aby uzyskać bardziej wiarygodne wyniki.

Okres: 01.01.2022 - 05.06.2023

Instrument: BTC/USD

Interwał: M15



- Jakość modelowania strategii wyniosła 33.85%, co jest niższym wynikiem niż w poprzedniej analizie.
- Całkowity zysk netto wzrósł do \$659.52, co stanowi poprawę w porównaniu do poprzedniego okresu.
- Maksymalna strata względna wyniosła 2.54%, wskazując większe ryzyko straty w porównaniu do wcześniejszego okresu.
- Odnotowano większą liczbę transakcji stratnych, które stanowiły 19.43% wszystkich transakcji.

- Największa strata bezwzględna wyniosła 239.80 jednostek waluty, wskazując na większe ryzyko straty.

Zmiana interwału czasowego

Po zmianie interwału czasowego z M15 na M30, a jednocześnie ustawienia symulacji na H1, nastąpiły pewne zmiany w raporcie:

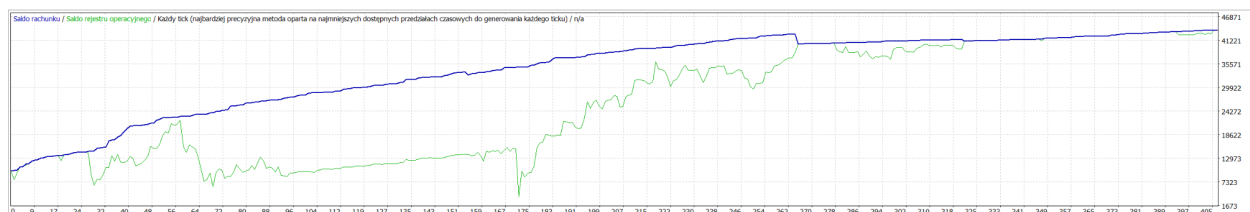
Słupki w teście	20531	Ticki użyte w modelu	115104140	Jakość modelowania	n/a
Błędy na wykresie	7580				
Depozyt początkowy	10000.00			Spread	Aktualny (2149)
Całkowity zysk netto	4449.74	Zysk brutto	5051.01	Strata brutto	-601.27
Wskaźnik zysku	8.40	Przewidywany zysk	15.08		
Największa strata bezwzględna	1796.72	Maksymalna strata względna	3635.61 (30.71%)	Strata relatywna	30.71% (3635.61)
Ilość transakcji w sumie	295	Krótkie pozycje (zyskowych %)	122 (88.36%)	Długie pozycje (zyskowych %)	173 (81.50%)
		Transakcje zyskowe (% wszystkich)	261 (88.47%)	Transakcje stratne (% wszystkich)	34 (11.53%)
	Największa	zyskowa transakcja	222.94	stratna transakcja	-367.15
	Średnia	zyskowa transakcja	19.35	stratna transakcja	-17.68
	Maksymalna	ilość transakcji zyskowych pod rząd (zysk w pln)	52 (711.32)	ilość transakcji stratnych pod rząd (straty w pln)	5 (-3.69)
	Największy	nieprzerwany zysk (suma transakcji zyskowych)	1244.08 (28)	nieprzerwana strata (suma transakcji stratnych)	-367.15 (1)
	Średnia	ilość transakcji zyskowych pod rząd	10	ilość transakcji stratnych pod rząd	1

- Liczba słupków w teście zmniejszyła się z 32 671 do 20 531, co wynika z większego interwału czasowego.
- Wprowadzenie większego interwału spowodowało również zwiększenie liczby ticków użytych w modelu z 37 331 297 do 115 104 140, co może wpływać na jakość modelowania.
- Całkowity zysk netto wzrósł z \$450.61 do \$4,449.74, co jest znaczącym wzrostem wyniku finansowego.
- Wskaźnik zysku również zwiększył się z 22.79% do 8.40%, co oznacza, że zyski są proporcjonalnie mniejsze w stosunku do strat.
- Maksymalna strata względna wynosi teraz 30.71%, co oznacza większe ryzyko straty w porównaniu do wcześniejszej konfiguracji.
- Wnioski te sugerują, że zmiana interwału czasowego na M30 i ustawienia symulacji na H1 miały istotny wpływ na wyniki handlowe. Zwiększyła się liczba transakcji, a wskaźnik zysku i efektywność handlowa nieco się zmniejszyły. Warto przeprowadzić dalszą analizę i optymalizację parametrów, aby znaleźć optymalne ustawienia dla strategii handlowej.

Optymalizacja parametrów

Podczas testowania różnych wartości parametrów uzyskano poniższy wynik dla **lot_size** = 0.65 i **min_gapsize** = 1.

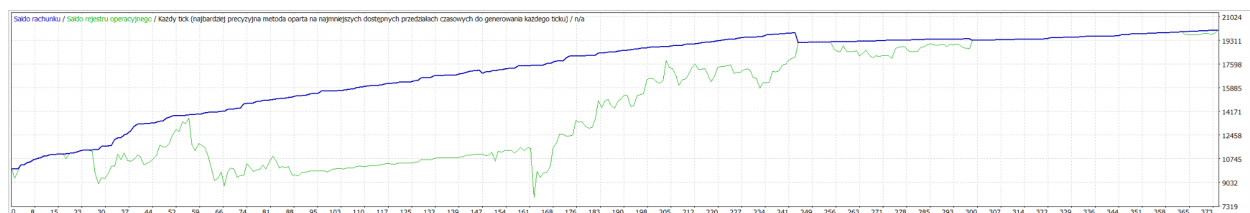
Słupki w teście	20531	Ticki użyte w modelu	115104140	Jakość modelowania	n/a
Błędy na wykresie	7580				
Depozyt początkowy	10000.00			Spread	Aktualny (646)
Całkowity zysk netto	33613.46	Zysk brutto	37330.25	Strata brutto	-3716.79
Wskaźnik zysku	10.04	Przewidywany zysk	82.79		
Największa strata bezwzględna	9559.92	Maksymalna strata względna	22211.99 (88.06%)	Strata relatywna	98.06% (22211.99)
Ilość transakcji w sumie	406	Krótkie pozycje (zyskowych %)	161 (88.76%)	Długie pozycje (zyskowych %)	245 (82.45%)
		Transakcje zyskowe (% wszystkich)	361 (88.92%)	Transakcje stratne (% wszystkich)	45 (11.08%)
	Największa	zyskowa transakcja	1468.60	stratna transakcja	-2375.87
	Średnia	zyskowa transakcja	103.41	stratna transakcja	-82.60
	Maksymalna	ilość transakcji zyskowych pod rząd (zysk w pln)	30 (3283.86)	ilość transakcji stratnych pod rząd (straty w pln)	3 (-79.27)
	Największy	nieprzerwany zysk (suma transakcji zyskowych)	6936.77 (23)	nieprzerwana strata (suma transakcji stratnych)	-2380.04 (2)
	Średnia	ilość transakcji zyskowych pod rząd	11	ilość transakcji stratnych pod rząd	1



Wnioski z raportu sugerują, że przy tych parametrach strategia handlowa osiągnęła wysoki wskaźnik zysku, generując znaczący zysk netto. Jednakże, należy zauważyć, że maksymalna strata względna wynosi 98.06%, co oznacza, że istnieje ryzyko ogromnych strat w porównaniu do zysków.

Warto również zwrócić uwagę, że strategia wykazała wysoki odsetek transakcji zyskownych (88.92%), a także relatywnie niską liczbę transakcji stratnych. To sugeruje, że parametry mogą sprzyjać generowaniu sygnałów handlowych, które często kończą się zyskownie.

Skłupi w testie	20531	Ticki użyte w modelu	115104140	Jakość modelowania	n/a
Błędy na wykresie	7580				
Depozyt początkowy	10000.00			Spread	Aktualny (915)
Całkowity zysk netto	10037.30	Zysk brutto	11201.97	Strata brutto	-1164.67
Wskaźnik zysku	9.62	Przewidywany zysk	26.84		
Największa strata bezwzględna	3070.15	Maksymalna strata względna	6919.72 (49.96%)	Strata relatywna	49.96% (6919.72)
Ilość transakcji w sumie	374	Krótkie pozycje (zyskownych %)	149 (86.66%)	Długie pozycje (zyskownych %)	225 (78.22%)
		Transakcje zyskowe (% wszystkich)	323 (86.36%)	Transakcje stratne (% wszystkich)	51 (13.64%)
		Transakcje zyskowe	450.80	Transakcje stratne	-731.61
		Średnia zyskowa transakcja	34.68	Średnia strata transakcja	-22.84
		Maksymalna ilość transakcji zyskowych pod rząd (zysk w pln)	23 (2115.51)	Maksymalna ilość transakcji stratnych pod rząd (strata w pln)	3 (-1.69)
		Największy nieprzerwany zysk (suma transakcji zyskowych)	2115.51 (23)	Największa nieprzerwana strata (suma transakcji stratnych)	-731.61 (1)
		Średnia ilość transakcji zyskowych pod rząd	7	Średnia ilość transakcji stratnych pod rząd	1



Ostatecznie zdecydowano się na **lot_size** = 0.2 i **min_gapsize** = 1 aby zmniejszyć ryzyko strat.

Pomysły rozwoju

- **Filtracja sygnałów:** Warto rozważyć dodanie dodatkowych warunków filtrujących, które będą musiały zostać spełnione przed otwarciem pozycji. Wykorzystanie dodatkowych wskaźników technicznych lub analizy sentymentu rynkowego może pomóc w eliminacji niepożądanych sygnałów i zwiększeniu trafności strategii.
- **Zarządzanie ryzykiem:** Należy skupić się na rozwinięciu lepszego zarządzania ryzykiem. Zaleca się wprowadzenie zasad stop-loss i take-profit, które będą chronić przed nadmiernymi stratami i pomogą zabezpieczyć osiągnięte zyski. Analiza stosunku ryzyka do nagrody oraz optymalizacja wielkości pozycji mogą przyczynić się do zrównoważonego zarządzania ryzykiem.
- **Analiza dodatkowych wskaźników:** W celu poprawy skuteczności strategii, warto przeprowadzić analizę dodatkowych wskaźników technicznych lub narzędzi analitycznych. Wykorzystanie różnych wskaźników trendu, oscylatorów lub formacji cenowych może dostarczyć dodatkowych informacji, umożliwiających trafniejsze generowanie sygnałów handlowych.

Finalny kod

Instrument: BTC/USD

Interwał: H1

```
extern int min_gapsize = 1;
extern double lotsize_gap = 0.2;
datetime order_time = 0;

int init() { return(0); }
int deinit() { return(0); }

int start() {
    // Defining the variables to decide.
    Print("order time", order_time);
    double current_openprice = iOpen(Symbol(), PERIOD_M30, 0);
    double previous_highprice = iHigh(Symbol(), PERIOD_M30, 1);
    double previous_lowprice = iLow(Symbol(), PERIOD_M30, 1);
    double point_gap = MarketInfo(Symbol(), MODE_POINT);
    int spread_gap = MarketInfo(Symbol(), MODE_SPREAD);
    datetime current_time = iTime(Symbol(), PERIOD_M30, 0);

    // Catching the gap on sell upper gap
    if (current_openprice > previous_highprice + (min_gapsize + spread_gap) * point_gap
        && current_time != order_time) {

        int ticket = OrderSend(Symbol(), OP_SELL, lotsize_gap, Bid, 0, 0,
                                previous_highprice + spread_gap*point_gap, "", 4, 0, Red);

        order_time = iTime(Symbol(), PERIOD_M30, 0);
        Print("I am inside (sell)", order_time);

        if (ticket < 0) {
            Print("OrderSend failed with error: ", GetLastError());
        }
    }

    // Catching the gap on buy down gap
    if (current_openprice < previous_lowprice - (min_gapsize + spread_gap) * point_gap
        && current_time != order_time) {
        ticket = OrderSend(Symbol(), OP_BUY, lotsize_gap, Ask, 0, 0,
                            previous_lowprice - spread_gap*point_gap, "", 5, 0, Green);

        order_time = iTime(Symbol(), PERIOD_M30, 0);
        Print("I am inside (buy)", order_time);
        if (ticket < 0) {
            Print("OrderSend failed with error #", GetLastError());
        }
    }
    return(0);
}
```