

Lab 2

1. Afișeze ultimele 10 bare de preț (Open, Close, High, Low) în ordine cronologică inversă (de la cele mai recente la cele mai vechi).

Trimită o ordine de vânzare (SELL) / (BUY pentru varianta 2) pe simbolul curent (de exemplu, EURUSD), cu un volum fix de 0.01 loturi.

Închidă automat poziția la dezactivarea EA-ului (OnDeinit).

Afișeze prețurile Ask și Bid la fiecare tick (OnTick).

```
Alert("Bar ", i, ": ", "Open: ", Open[i], ", Close: ", Close[i], ", High: ", High[i], ", Low: ", Low[i]);
```

```
const float VOLUME_TO_SELL = 0.01;  
orderTicket = OrderSend(Symbol(), OP_SELL, VOLUME_TO_SELL, Bid, 0, 0, 0);
```

```
OrderClose(orderTicket, NUMBER_OF_LOTS, Ask, SLEEPAGE);
```

```
// Note: the buying price (ask) in global variable, does not need to be  
orderTicket = OrderSend(Symbol(), OP_BUY, VOLUME_TO_BUY, Ask, 0, 0, 0);
```

```
OrderClose(orderTicket, NUMBER_OF_LOTS, Bid, SLEEPAGE);
```

2. La pornire, trimite aleatoriu o comandă de BUY sau SELL

Dacă diferența absolută între prețul de deschidere și prețul curent depășește 0.001 (10 pip), atunci:

- Închide comanda deschisă
- Trimite o comandă nouă de tip opus (dacă era BUY, devine SELL și invers)

```
const int RANDOM = MathRand();
```

```
(MathAbs(operationStartPrice - Ask) > TRESHOLD)
```

Lab 3

3. Analizează prețurile de închidere ale ultimelor 10 bare

Dacă prețurile cresc consecutiv:

- Închide orice poziții de cumpărare deschise
- Deschide o poziție de vânzare

Dacă prețurile scad consecutiv:

- Închide orice poziții de vânzare deschise

- Deschide o poziție de cumpărare

```
if (buyOrderTicket != -1 && OrderSelect(buyOrderTicket, SELECT_BY_TICKET))
{
    OrderClose(buyOrderTicket, VOLUM, Bid, 0);
    buyOrderTicket = -1;
}
```

Lab 4

4. La apariția unei noi bare:

Dacă ultimele N bare au Close < Open (bări roșii/descrescătoare):

- Închide orice poziții de vânzare existente
- Deschide o poziție de cumpărare

Dacă ultimele N bare au Close > Open (bări verzi/crescătoare):

- Închide orice poziții de cumpărare existente
- Deschide o poziție de vânzare

```
long currentCallTime = Time[1];
```

```
bool hasNewBarAppear()
{
    // For this we can use the Time[] array, which contains the time of the last bar
    static long lastCallTime = 0; // 0 seconds
    long currentCallTime = Time[1];

    if (currentCallTime != lastCallTime)
    {
        // Reset the time of the last function call
        lastCallTime = currentCallTime;
        return true;
    }
    else
        return false;
}
```

Lab 5

5. La pornire:

- Deschide aleatoriu o poziție de Cumpărare (BUY) sau Vânzare (SELL)

În timpul funcționării:

- Dacă prețul crește cu X puncte față de ultima poziție de Cumpărare → deschide o nouă Cumpărare
- Dacă prețul scade cu X puncte față de ultima poziție de Vânzare → deschide o nouă Vânzare

Gestionarea ordinelor:

- Fiecare ordin nou are stop-loss și take-profit setate la o distanță fixă (DISTANCE)
- Memorează prețurile ultimelor ordine în variabile globale

```
// Check if buy price is higher than average
if (isBuyPriceAboveAverage(lastBarsCount, numberOfPoints))
{
    // stop loss: we assure that the order does not loose more money than "stopLoss"
    double STOP_LOSS = Ask - sltpDistance * Point;

    // We assure that the profit equal to "takeProfit" is not lost
    double TAKE_PROFIT = Ask + sltpDistance * Point;

    // And start buying a new buy
    OrderSend(Symbol(), OP_BUY, VOLUME_TO_TRANSACT, Ask, 0, STOP_LOSS, TAKE_PROFIT);
}
// Else, check if the sell price is lower than average
else if(isSellPriceBelowAverage(lastBarsCount, numberOfPoints))
{
    // stop loss: we assure that the order does not loose more money than "stopLoss"
    double STOP_LOSS = Bid + sltpDistance * Point;

    // We assure that the profit equal to "takeProfit" is not lost
    double TAKE_PROFIT = Bid - sltpDistance * Point;

    // and start selling
    OrderSend(Symbol(), OP_SELL, VOLUME_TO_TRANSACT, Bid, 0, STOP_LOSS, TAKE_PROFIT);
}
```

Lab 6

6. Dacă prețul curent (Ask/Bid) depășește media ultimelor N bare (fără bara curentă) cu mai mult de X puncte, atunci:

- Se trimite o comandă de tip BUY, cu Stop Loss și Take Profit la o distanță de $y = x$ puncte.

Dacă prețul curent scade cu mai mult de X puncte sub această medie, atunci:

- Se trimite o comandă de tip SELL, cu Stop Loss și Take Profit la aceeași distanță $y = x$ puncte.

Media se calculează ca fiind media aritmetică a punctelor de mijloc ($\text{High} + \text{Low}$) / 2 pentru ultimele N bare (fără bara curentă).

Verificările se fac doar când apare o bară nouă, nu la fiecare tick.

Lab 8

7. Expert Advisor-ul (EA) trebuie să monitorizeze profitul total al tuturor ordinelor deschise și să le închidă automat atunci când:

- Profitul total depășește o sumă stabilită (+S USD) SAU
- Pierdere totală depășește aceeași sumă (-S USD)

8. La apariția fiecărei bare noi (nu la fiecare tick):

- Calculează media ($\text{High} + \text{Low}$) / 2 a ultimelor N bare (fără bara curentă).
- Dacă prețul Ask este cu X puncte peste media calculată, se trimite un ordin Buy:
 1. Cu volum prestabilit.
 2. Cu StopLoss și TakeProfit la distanța de $Y = X$ puncte față de prețul de cumpărare.
- Dacă prețul Bid este cu X puncte sub media calculată, se trimite un ordin Sell:
 1. Cu aceleași condiții de volum, StopLoss și TakeProfit.

La fiecare tick, verifică:

- Dacă profitul total al ordinelor deschise este mai mare sau egal cu o valoare +S (ex: +10 USD), sau mai mic sau egal cu -S (ex: -10 USD), atunci închide toate ordinele.

9. Funcția `getBuyOrderAskPrice()` are ca scop determinarea celui mai recent preț de deschidere (Ask) dintre ordinele BUY deschise și închise, apoi întoarce cel mai mare dintre ele. Dar codul are niște ambiguități și greșeli pe care le voi evidenția la final. -> lab 5

Lab 9

10. La inițializare (OnInit):

- Dacă nu există niciun ordin deschis, se trimite un ordin nou (BUY sau SELL, ales aleator), cu volum 0.01.

La fiecare tick (OnTick):

- Se verifică dacă există un ordin deschis:
 1. Dacă da: nu se face nimic.
 2. Dacă nu:
 1. Se ia ultimul ordin închis.
 1. Dacă a fost profitabil: se trimite un ordin nou (BUY sau SELL aleator), cu volum 0.01.
 2. Dacă a fost în pierdere: se trimite un ordin nou, de tip opus față de cel anterior, dar cu volum dublu față de ultimul ordin închis.

Fiecare ordin trimis trebuie să aibă:

- Stop Loss și Take Profit la o distanță fixă x, configurabilă.
- Un magic number propriu (unic), și codul EA-ului trebuie să gestioneze doar ordinele cu acest magicNumber.

```
extern double MinVol = 0.01;
extern int MagicNumber = 1234;
extern int SLTPDist = 20;

bool OpenOrderExist()
{
    for (int i=OrdersTotal()-1;i>=0;i--)
        if (OrderSelect(i,SELECT_BY_POS))
            if (OrderMagicNumber()==MagicNumber)
                return true;
    return false;
}

int OnInit()
{
    if (OpenOrderExist())
        return INIT_SUCCEEDED;

    if (MathRand()%2==0)
        OrderSend(Symbol(), OP_BUY, MinVol, Ask, 0, Ask-SLTPDist*Point, Ask+SLTPDist*Point, NULL, MagicNumber);
    else
        OrderSend(Symbol(), OP_SELL, MinVol, Bid, 0, Ask+SLTPDist*Point, Ask-SLTPDist*Point, NULL, MagicNumber);
    return(INIT_SUCCEEDED);
}
```

// Important! MODE_HISTORY gives only closed orders, so we can be sure these orders are closed

```
void SelectLastClosedOrder()
{
    for (int i=OrdersHistoryTotal()-1;i>=0;i--)
        //Selectează ultimul ordin închis cu MagicNumber-ul nostru.
        if (OrderSelect(i,SELECT_BY_POS,MODE_HISTORY))
            if (OrderMagicNumber()==MagicNumber)
                return;
}
```

Lab 10

11. Scrie un Expert Advisor care:

- La inițializare (OnInit) plasează o rețea (grid) de ordine de tip Buy Limit și Buy Stop, deasupra și dedesubtul prețului curent.
- Plasează n ordine Buy Stop și n ordine Buy Limit, la distanță $x * \text{Point}$ între ele.
- Fiecare ordin are Take Profit și Stop Loss la distanță $y * \text{Point}$.
- Toate ordinele (deschise sau în așteptare) au același MagicNumber pentru a putea fi recunoscute.
- În OnTick, monitorizează toate ordinele deschise (cu acel MagicNumber) și calculează profitul total.
- Dacă profitul total atinge pragul z (pozitiv sau negativ), atunci:
 1. închide toate ordinele deschise;
 2. șterge toate ordinele în așteptare (Buy Limit / Buy Stop);