INSTRUCTIONS FOR SESSION 8 TASK

5% of your semester grade
Time required: 3 hours
Suggestion: Print this document out before trying the activity

PREPARATION

1. Create a new Microsoft Word document (use WordPad if you do not have MS Word).
2. Save your empty document to your desktop with the filename *WillRennerSession8.doc* (use your own name instead of my name). Leave the file open while doing the activities below.

NB The following activities require you to make screen shots of your AICE codes, and then paste these into the word document which you created above. Screenshots from the remote computer can be done by holding down ALT and then clicking the *PrintScreen* key on the keyboard. To quickly paste the screenshot into the word document on your local computer, just right click in the document and select paste. To compress screen shots, save them as .jpg images from within the Paint application.

Activity 1 (3 marks)

1.1 Please create a short ALICE script that will store brokers that have made tick movements to securities into an array indexed by security and number. Please take a screenshot of your ALICE source code and paste it into the word document. (1 mark)

Security (index)	Number (index)	Broker (value)
CVC	1	X
CVC	2	Y
BHP	1	Y
ABC	1	Z

Your task for 1.1 is to populate an array of brokers indexed by security and number which is populated with data (such as that shown above), where:

- The broker + security pair is unique (meaning that each broker+security combination can only appear once in the array.
- The number indicates the occurrence order of a broker making movements to a security; where ‘movement to a security’ is defined as trades which differ in price from the last trading price.

Your script should look like this:

```
declare BrokerSecurity_Pair_Seen[security, house] : boolean
declare NumOfBrokers_For_Security[security] : number
declare Broker_For_Security[security, number] : house

on trade
        if price > lastprice
                and undefined(BrokerSecurity_Pair_Seen[security, buyerh])
        then
                BrokerSecurity_Pair_Seen[security, buyerh] = [redacted]
                NumOfBrokers_For_Security[security] += 1
                Broker_For_Security[security, [redacted]] = [redacted]
        end if

        if price < lastprice
                and undefined(BrokerSecurity_Pair_Seen[security, sellerh])
        then
                [redacted]
        end if
end on
```

Please complete the above code by filling in the hidden parts.

Note, that for positive price movements, we need to store the buyer house, and for negative price movements, we store the seller house. The variable *Broker_For_Security* stores the list of brokers making movements to a security. The logic is similar to Activity 2 from Session 6.

1.2 Please add your script from 1.1 to the following script appropriately in order to create a list of brokers making movements to securities since 3 trading days ago. Please take a screenshot of your ALICE code and paste it into the word document. (1 mark)

```
declare Tick_Movement_For_Broker[security, house, string] : number
on trade
        if price > lastprice then
                Tick_Movement_For_Broker[security, buyerh, "UP"] = tickdiff(price, lastprice)
        elsif price < lastprice then
                Tick_Movement_For_Broker[security, sellerh, "DOWN"] = abs(tickdiff(price, lastprice))
        end if
end on

on info
        IF PRICE MOVES UPWARDS
                FIND OUT BROKER X WITH THE BIGGEST Tick_Movement_For_Broker[security, house, "UP"]
        IF PRICE MOVES DOWNWARDS
                FIND OUT BROKER X WITH THE BIGGEST Tick_Movement_For_Broker[security, house, "DOWN"]
        END IF

        ALERTING ..... ,
        house = BROKER X
end on

Benchmarks_below: 3 days
declare Tick_Movement_For_Broker[security, house, string] : number
on trade
        if price > lastprice then
                Tick_Movement_For_Broker[security, buyerh, "UP"] = tickdiff(price, lastprice)
        elsif price < lastprice then
                Tick_Movement_For_Broker[security, sellerh, "DOWN"] = abs(tickdiff(price, lastprice))
        end if
end on
```

1.3 Based on your solution to 1.2, create a for loop that will loop over the broker lists created in 1.2 and find out the broker making the highest tick movements on the suspect security since 3 trading days ago and up to the point when the insider trading alert is triggered. This broker

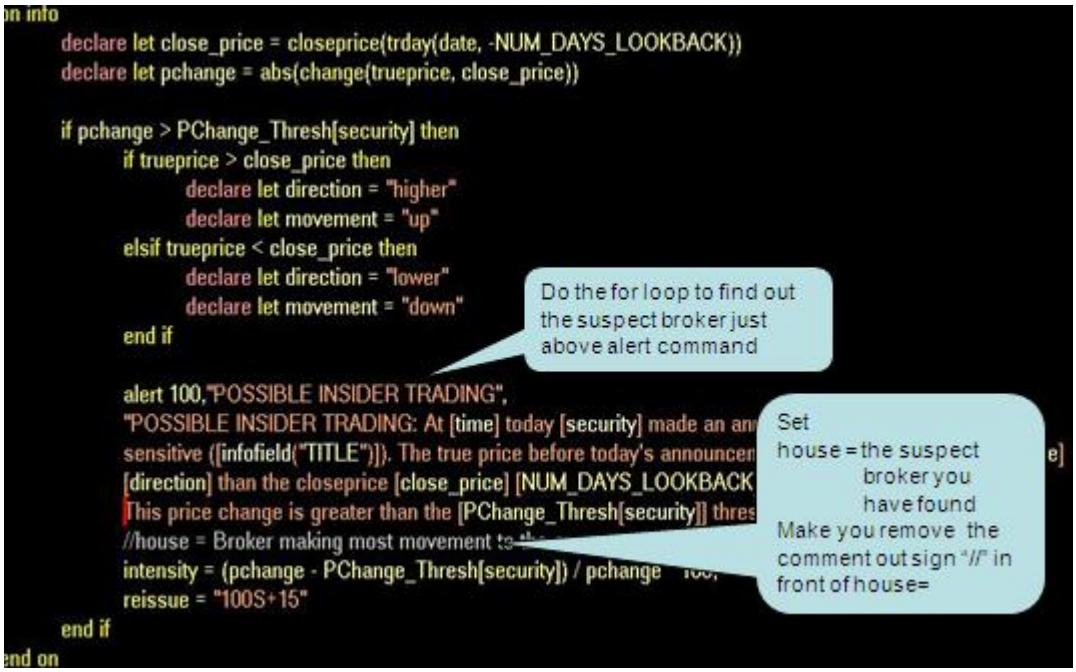
becomes the suspect broker. Please set the *house parameter* to be the suspect broker for the insider trading alert. Please take a screenshot of your ALICE source code and paste it into the word document. (1 mark).

Hints:

a) If the insider trading alert is triggered for abnormal upward price movements, then the suspect broker must be making the highest tick movements to the suspect security and the vice versa.

b) It may occur to you what will happen if there are more than 1 broker making the same tick movements to the suspect security and that amount of movements is the highest among all brokers. In this case, you only need to set the first broker making the highest movements from your list to be the suspect broker.

c) The following screenshot may be helpful to you:



Activity 2 (2 marks)

In the lecture, we learnt how to create databases for Market Orders and Limit Orders. Now let's replicate the same procedures for Limit Order Cancellations.

2.1 Create an ALICE script that will print the following field of limit order cancellations into a csv file: number index, date, time security, price, volume and broker. Please take a screenshot of your ALICE source code and paste it into the word document. (1 mark)

2.2 Now create an ALICE script that will read in the csv file you created for 2.1 and print the total number of limit order cancellations and average volume per order cancellation into a csv file. Please take a screenshot of your ALICE source code and paste it into the word document. (1 mark)