Introduction:

The first step in automated trading is to process a stream of market data that is disseminated by the exchange. This short quiz will simulate processing such feed (with the exception that the raw data is read from file vs from a socket).

The information stream contains two types of data:

a. Information about quotes of different symbols in the market, for example AAPL is now quoted at $45.23 bid (i.e. to buy) and $45.27 ask (i.e. to sell)
b. Information about trades in the market, for example 100 shares of AAPL were just traded at $45.23.

In this exercise you are asked to read such raw information as it has been captured in a file, and print out the trades that were captured during that time.

Specifications:

The input file - input_test.dat - is attached here.

The file contains packets of market data starting with the first byte of the file. All data is in network byte order.

Each packet has the following header:

Name	Offset	Num Bytes	Type	Description
Packet Length	0	2	Integer	Length of the entire packet (Including this data member).
Num Market Updates	2	2	Integer	Number of market updates in this packet.

The body of the packet (immediately after the header) contains one or more market updates. Each market update is either a quote or a trade.

Quote data has the following format:

Name	Offset	Num Bytes	Type	Description
Length	0	2	Integer	Length of the entire market update (Including this data member).
Type	2	1	Alpha	"Q"
Symbol	3	5	Alpha	Left-justified string with spaces filling the unused bytes.
Price Level	8	2	Integer	Zero-indexed price level to apply this update.
Price Level Size	10	2	Integer	New size of price level (given in round lots; 100's).
Price Level Price	12	8	Integer	New price of price level (see below*).
Dynamic Data	20	Dynamic	Alpha	Reserved data; can be ignored.

Trade data has the following format:

Name	Offset	Num Bytes	Type	Description
Length	0	2	Integer	Length of the entire market update (Including this data member).
Type	2	1	Alpha	"T"
Symbol	3	5	Alpha	Left-justified string with spaces filling the unused bytes
Trade Size	8	2	Integer	Volume traded (given in round lots; 100's).
Trade Price	10	8	Integer	Price of traded volume (see below*).
Dynamic Data	18	Dynamic	Alpha	Reserved data; can be ignored.

*Prices are fixed point decimal numbers equal to the product of 10000 and the actual price.

The Requirements:

Read in the custom market data from file. Print trade data to a text file using the printf-style format string:

```
"%d %s @ %.2f", size, symbol, price
```

Ignore quote data.

The first two lines of your output will be the following:

17000 CAT @ 4.72
70600 AA @ 6.45

In reply send all your C++ files and your output file.

Important notes:

1. You may assume that this is the only file that this program will ever process. There is no need to make the program generic to process other files, so feel free to hard-code whatever you like.
2. Minimize the number of file read operations. A solution that will read a byte or an integer at a time is not acceptable (and not feasible in real life). Also, eliminate any unnecessary memory moving or copying.
3. The ideal approach is to use a struct to interpret the data read from file.