**股票交易数据处理大作业任务**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date** | **Open** | **High** | **Low** | **Close** | **Adj Close** | **Volume** | **Code** | **Change** | **Pct\_change** |
| 2017-01-03 | 9.1 | 9.18 | 9.09 | 9.16 | 9.028069 | 45984049 | 000001.SZ | 0.06 | 0.0065934 |
| 2017-01-04 | 9.15 | 9.18 | 9.14 | 9.16 | 9.028069 | 44932953 | 000001.SZ | 0 | 0 |
| 2017-01-05 | 9.17 | 9.18 | 9.15 | 9.17 | 9.037926 | 34437291 | 000001.SZ | 0.01 | 0.0010917 |
| 2017-01-06 | 9.17 | 9.17 | 9.11 | 9.13 | 8.998502 | 35815420 | 000001.SZ | -0.04 | -0.004362 |
| 2017-01-09 | 9.13 | 9.17 | 9.11 | 9.15 | 9.018213 | 36108157 | 000001.SZ | 0.02 | 0.0021906 |
| 2017-01-10 | 9.15 | 9.16 | 9.14 | 9.15 | 9.018213 | 24105395 | 000001.SZ | 0 | 0 |

如上表所示，现有某个阶段中国股市的cvs格式数据（数据样例如上，从左开始的列名依次是日期、开盘价、最高价、最低价、收盘价、调整后收盘价、成交量、股票代码、价格上涨、价格上涨百分比），请你编写程序对其进行分析处理并完成以下任务。

**一、读入数据**

实现函数LoadStockData，并对补齐缺失数据，具体要求如下：

* 加载cvs格式的股票数据，保存在适当的数据结构中。
* 加载成功，返回总的记录条数；否则返回-1
* 数据中有些缺失（例如，某只股票某天停盘会缺失开盘价、最高价、最低价、收盘价等信息）
* 对于这些有缺失的数据，请将所有价格（包括开盘、收盘等等）用数据缺失前离这个数据最近“调整后收盘价(Adj Close)”调整后收盘价替代。LoadStockData函数原型如下：

int **LoadStockData**(char \*filename);

**二、查询类任务**

**1、查询某只股票的信息，请实现以下Query函数**

int **Query**(char \* para\_str );

函数说明：

1）函数参数 para\_str 表示查询的参数语句，具体格式如下：

**-C “code”-d “date”** -o -c -a -h -l

其中：

-C后面跟一个股票代码，形如：-C 000001.SZ；

-d后面跟一个具体的日期，例如：-d 2017-01-03；

-o 表示查询开盘价；

-c 表示查询收盘价；

-a 表示查询调整后的收盘价；

-h 表示查询最高价；

-l 表示查询最低价。

**注意事项：**

* 一个查询语句中一定包含查询参数C和 -d，其他查询参数-o, -c, -a, -h, -l至少出现一个。
* 查询参数的顺序可随意, 例如 -C 000001.SZ -h -d 2017-01-03。

2）函数返回值：查询失败时，该函数的返回-1；查询成功时，返回0，并需要向标准输出设备输出一行查询结果，输出的格式如下：

一行，以code date开头，code表示查询的股票代码，date表示查询的日期；后面按照查询语句中的信息代码顺序输出查询参数和查询的结果。

例如：Query(“C 000001.SZ -h -d 2017-01-03 -a”)表示查询股票000001.SZ 2017年1月3日的最高价和调整后收盘价，查询结果输出应为：000001.SZ 2017-01-03 -h 3.18 -a 9.02

**2、按日期的Top K查询，即查询复合条件的前（后）k条记录**。请实现下面的函数：

int **TopK**(char \*date, char \*data, int k, int desc);

函数说明：

1）函数参数：

date表示日期；

data表示查询的数据指标，可能为取值为”Volumn”、”Range”、”AdjClose”，分别对应查询成交量、波动（考虑股票价格波动值（即最高价–最低价）与开盘价百分比）、调整后收盘价的Top k。

k：表示要查询前k项或者后k项（根据第4个参数desc决定是前k项还是后k项），

desc: 0或者1，0表示查询前k项，1表示查询后k项。

2）函数返回值：查询失败时，函数返回-1；查询成功时，函数返回值为符合条件的记录条数，并同时按照顺序输出查到的k条记录，每条记录输出的格式如下：

**股票代码 日期 开盘价 收盘价 调整后收盘价 股票价格波动百分比 成交量**

**注意事项：**

* 有小数的，保留2位小数（调整后收盘价保留6位小数）
* 如果2条记录的查询指标（成交量、波动、调整后收盘价）一样，则股票代码小的排在前面。

例如：TopK(“2017-01-03”, “Volumn”, 3, 1) 是查询2017年1月3日成交量最小3支股票

**三、计算类任务**

**1、股票数据的基本统计指标**，统计计算某只股票在给定时间段内某项数据指标的均值或标准差。

所需的函数形式为 double **Calculate**(char \*para\_str);

函数说明：

1）函数参数 para\_str 表示统计计算的参数语句，具体格式类似股票基本信息查询中的规定：

**-C “code”-d “start\_date” -d “end\_date”** -o -v

其中：

-C后面跟一个股票代码；

-d后面跟一个具体的日期，第1个表示开始日期，第2个表示结束日期

-o 表示开盘价；

-c 表示收盘价；

-h 表示最高价；

-l 表示最低价；

-v 表示计算平均值；

-s 表示计算标准偏差， 其计算公式采用

https://gss2.bdstatic.com/9fo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D150/sign=49a3c31e8f1001e94a3c100a880f7b06/d058ccbf6c81800af3b703f9b33533fa838b47f3.jpg, 其中*x*i表示数据项，*μ*表示均值。

**注意事项：**

* 指定股票代码及起止日期的参数为固定参数选项；
* 四个关于数据字段的参数 -o, -c, -h, -l 为其中之一；
* 计算指标参数 -v -s 为二选一。各类参数的先后顺序保持不变。

2）函数返回值：如果查询统计成功，返回一个计算所得的双精度数，并保留4位小数，输出到标准设备；如果失败，返回-1.0，并输出-1。

**2、股票分析技术指标**，计算某只股票在指定日期范围内的MACD指标 [1]

MACD指数平滑移动平均线(Moving Average Convergence and Divergence)又称指数离差指标，是移动平均线原理的进一步发展。这一技术分析工具自1971年由查拉尔德拉徘尔创造出来之后，一直深受股市投资者的欢迎。MACD的原理是运用短期(快速)和长期(慢速)移动平均线聚合和分散的征兆加以双重平滑运算，用来研判买进与卖出的时机，在股市中这一指标有较大的实际意义。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Code** | **Date** | **Close** | **Volume** | **EMA\_12** | **EMA\_26** | **DIF** | **DEA** | **MACD** |
| 002765.SZ | 2016-07-21 | 17.02 | 144470645 | 18.01466 | 18.12856 | -0.1139 | 0.376717 | -0.98124 |
| 002765.SZ | 2016-07-22 | 17.39 | 233970928 | 17.91856 | 18.07385 | -0.1553 | 0.270315 | -0.85122 |
| 002765.SZ | 2016-07-25 | 17.7 | 235433919 | 17.88493 | 18.04616 | -0.16123 | 0.184007 | -0.69047 |
| 002765.SZ | 2016-07-26 | 17.78 | 149319656 | 17.86879 | 18.02645 | -0.15765 | 0.115674 | -0.54666 |
| 002765.SZ | 2016-07-27 | 17.42 | 147648626 | 17.79975 | 17.98152 | -0.18178 | 0.056184 | -0.47592 |

如上表所示，要得到MACD指标，需要分别计算如下指标：

1）长、短周期对应的每日指数平均数EMA\_N(i)

EMA\_N(i) = P\*2/(n+1) + EMA(i-1)\*(n-1)/(n+1)，其中P表示当日收盘价，n为移动平均线周期的天数。常用的长、短周期的天数分别用26、12天。

2）离差值DIF(i)， 为短周期的EMA减去长周期的EMA, 比如：

DIF(i) = EMA\_12 (i) - EMA\_26(i)

3）离差值的M天移动平均值DEA，M参数常用9天，则：

DEA(i) = DEA(i-1) × 0.8 + DIF(i) × 0.2

4）计算MACD值时，可以采取不同的方式，一种是MACD(i) = (DIF(i) - MACD(i-1)) × 0.2 + MACD(i-1)；另一种是 MACD(i) = 2 × (DIF(i) - DEA(i))。本次作业如上表示例采用第二种。

注：新股上市第一天，其DIF、DEA以及MACD都为0，因为当天不存在前一天，无法做迭代运算，计算新股上市第二天的EMA时，第一天的EMA需要用收盘价（而非0）来计算。本作业不必从上市第1天开始计算，在给定日期范围之前一段时间（本题可假定在给定日期范围之前30天才上市）开始迭代计算即可。

现需要编写实现如下函数，计算股票的MACD指标：

void **CalcMacd**(char \*code, char \*start\_date, char \*end\_date);

函数说明：

1）函数参数： code表示股票的代码，start\_date表示开始日期， end\_date 表示结束日期。

2）函数返回值：函数无返回值。按行输出如上表所示的从指定日期开始的计算过程，数据项之间以空格分隔，数据项的顺序不变，浮点数除股价外(2位小数)，保留4位输出。

**注意事项：**

* 本函数可进一步自定义调用的其它子函数形式；
* 输出计算过程中的数据时，无需输出标题行。

**3、股票筛选**

**3.1 按技术指标筛选股票**

通过技术指标，如前面介绍的DMA和MACD等指标，进行买入和卖出的判断，是股市投资的一种重要手段。本小节介绍“金叉买入，死叉卖出”这种简单粗暴的追涨杀跌的投资策略，其基本思想是基于之前介绍的MACD技术指标。



如上图所示：白线表示MACD指标中的离差值DIFF、黄线代表DEA，即离差值的M天移动平均值DEA。白线上穿黄线为金叉（如图中从左往右的第一个和第三个叉），白线下穿黄线为死叉（如图中的中间的叉）。所谓“金叉买入，死叉卖出”就是在金叉点的时候买入股票，在“死叉”点的时候卖出股票。

现请你编写函数针对某一明确时间段（起始时间到终止时间的闭区间），计算出所有股票金叉点和死叉点的个数，并按照金叉点越多约优先；金叉点相同，死叉点越少越优先的顺序；如果金叉点和死叉点相同，则按照股票代码词典序列越低越优先，进行排序，并输出前k只股票（不满k只的情况，输出所有记录）。

int **MACDTopK**(char \*start\_date, char \*end\_date, int k);

函数说明：

1）函数参数：

• start\_date与end\_date分别表示起始与终止日期（考虑闭区间）。

• k：表示要查询前k条记录 。

2）函数返回值为符合条件的记录条数；并同时按照顺序输出查到记录，每条记录输出的格式如下：

股票代码 起始日期 终止日期 金叉点个数 死叉点个数

例如：MACDTopK(“2017-01-01”, “2017-01-31”, 3) 是查询2017年1月1日到1月31日按照金叉/死叉点排序规则下的最靠前的3条记录。

**3.2 股票代码模糊匹配**

股票代码是由六位数字和后缀组成的字符串，如“000001.SZ”，用户通过股票代码进行筛选时，很多时候因为记错或是不小心输错，而导致查询的代码不完全准确。例如，希望查找“000001.SZ”，却不小心输入了代码”00001.SZ”（少输了一个0）。此时，一个用户友好的系统不应该仅仅告诉用户没有找到股票，而是返回一些“相似”的股票代码作为推荐，提示用户他可能想输入的是这些“相似”的股票，从而方便用户纠错。

一种常见的找“相似”的办法是“编辑距离”函数。该函数的思想是度量将一个字符串变成另一个字符串所需要的最少操作数。它首先定义了三种基本编辑操作：

• 插入：在任意位置插入一个字符，如将00001.SZ变为000001.SZ，只需在前者的第1个位置插入一个字符0

• 删除：删除任意一个字符，如将0000011.SZ变为000001.SZ只需删除前者的第7个字符

• 替换：替换任意一个字符，如将000031.SZ变为000001.SZ只需将前者的第5个字符由3替换为0即可。

假设给你一个股票代码query作为查询，一个数字threshold作为允许的最大编辑距离数，请筛选出所有与query的编辑距离小于等于threshold的股票代码，作为模糊匹配的结果，并按照股票代码的词典序排序。

int **FuzzyMatch** (char \*query, int threshold);

函数说明：

1）函数参数：

• query表示查询的股票代码。

• th：表示允许的最大编辑距离阈值。

2）函数返回值为符合条件的股票代码个数；并同时按照词典序输出匹配到股票代码，每个代码输出一行。

例如：FuzzyMatch(“000011.SC”, 3) 是匹配与查询代码000011.SC的编辑距离小于等于3的所有股票代码，并按照词典序排序输出。

**四、开放性任务 （二选一）**

**1 、模拟交易**

假定现有入市启动资金100万RMB，在你具备了股市交易知识和投资经验的基础上，模拟一定时期内的股票买入卖出交易，以期获得较好的收益。虽然已经提供了一定时期内的沪深股市历史交易数据，但显然，本题设定的模拟交易的过程不是一个事后的寻优搜索问题，只能是用来计算、验证你所采取的交易策略的结果。

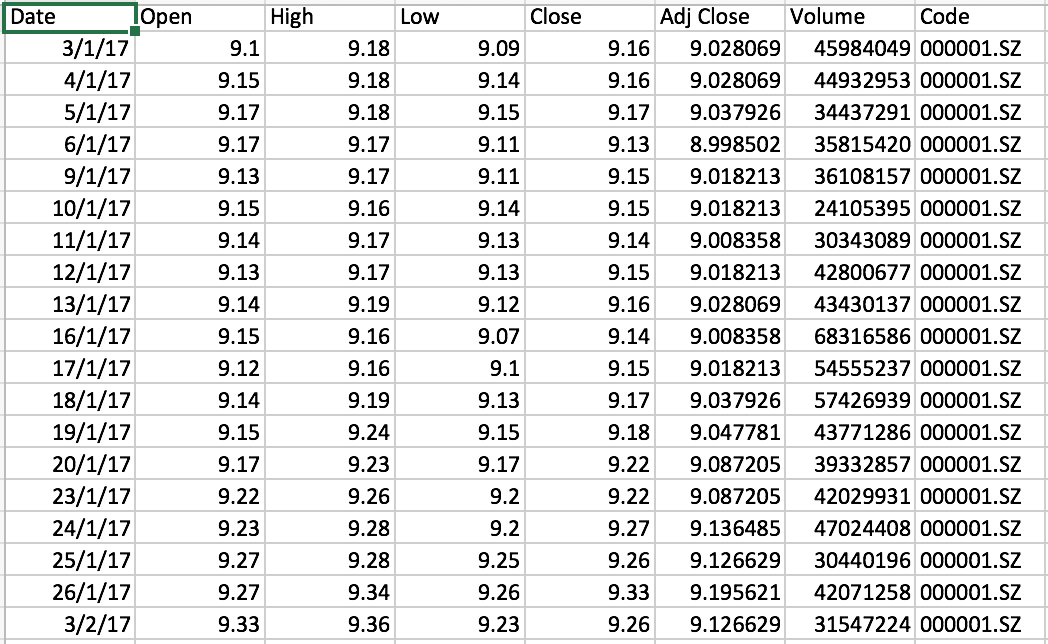
你需要根据你的“入市炒股经验”进行决策，比如考虑1）选哪些股；2）持股比例；3）减持换股或增持；4）全部卖出；5）重新入市等等交易时机和操作行为问题。

已知的股市交易股则有：1）当天买入股票不能卖出；2）交易需支付手续费用。虽然费用包含不同种类，为简化问题，我们统一规定为成交金额的0.3%，最低5元。

现给定一个入市日期，按自己设计的策略进行交易模拟，计算1年后所持股票的市值（现金账户余额），并与大盘的总体涨跌情况进行比较。你需要输出整个交易过程中买入卖出的交易记录。买入时，以每天的开盘价为准；卖出时，以收盘价为准。

**2、股票涨跌预测**

股票的涨跌预测是股市投资中的一个重要问题。现请你根据一个历史时间段内股市的整体情况，预测某只股票的涨跌。具体来说，给定某个日期date，请你根据date之前股票的记录信息，预测日期date+1某只股票的收盘价相比于日期date是涨还是跌。

一种可行的思路是利用中学接触过的最小二乘法。例如，针对下图中股票000001.SZ，我们可以考虑2017年2月3日往前一个月的历史数据，去估计2017年2月4日相对于2月3日的涨跌。

通过最小二乘法的思想，我们可以首先计算1月3日至2月3日这一阶段每天的收益，即log(当日收盘价)-log(前一日收盘价)，并将这些散点绘制在图上；进而使用最小二乘法拟合出一条曲线。并使用这条曲线预测2月4日的收益；如果为正，则预测涨；否则预测跌。

除了最小二乘法，你还可以考虑逻辑回归、Lasso回归等回归算法；此外，还可以同时考虑股市中其它相关股票的涨跌，来预测此只股票涨跌。

此任务的评测标准如下：提供给你一个日期date和date之前的一个时间段，让你预测某只股票在date+1日期的涨跌，即输出1表示涨；输出-1表示跌。我们会将你的预测结果与date+1日实际的涨跌对比：如果一致，则该只股票预测正确，否则错误。我们将所有股票预测的结果综合考虑，则得到预测方法的准确率（预测正确的股票占所有股票的比例）。

**参考**

[1] MACD：

<https://uqer.io/community/share/5799b908228e5ba291060674>

<http://www.360doc.com/content/14/1106/07/3033201_422867044.shtml>