字串及檔案的處理

字元及字串
課堂練習:計算生命數字
檔案處理
課堂練習:檔案輸入輸出
日期格式的轉換及天數的計算
使用文字檔和Excel交換資料
葛蘭碧法則驗證
課堂練習:葛蘭碧法則驗證
標的物歷史報酬及歷史波動度的估計

字元型態

- c語言的字元是用 char宣告
  - Ex: char a;
- 每個字元都用一個介於0~255整數表示,使用不同的方式解釋,就有不同意義
- ASCII表載明其對應(見下張投影片)
- 範例程式(可用ASCII表推算) 參見CharString Project

```
char a=33;
char b='B';
printf("%d %c\n",a,a);
printf("%d %c\n",b,b);
a=a+b;
printf("%d %c\n",a,a);
```

用單引號包住

ASCII表

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(nul)	0	0000	0x00	(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
(soh)	1	0001	0x01	!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
(stx)	2	0002	0x02	"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
(etx)	3	0003	0x03	#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
(eot)	4	0004	0x04	$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
(enq)	5	0005	0x05	%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
(ack)	6	0006	0x06	&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
(bel)	7	0007	0x07	'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(bs)	8	0010	0x08	(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
(ht)	9	0011	0x09	)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
(nl)	10	0012	0x0a	*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
(vt)	11	0013	0x0b	+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
(np)	12	0014	0x0c	,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
(cr)	13	0015	0x0d	-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
(so)	14	0016	0x0e	.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
(si)	15	0017	0x0f	/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
(dle)	16	0020	0x10	0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
(dc1)	17	0021	0x11	1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
(dc2)	18	0022	0x12	2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
(dc3)	19	0023	0x13	3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
(dc4)	20	0024	0x14	4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
(nak)	21	0025	0x15	5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
(syn)	22	0026	0x16	6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
(etb)	23	0027	0x17	7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
(can)	24	0030	0x18	8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
(em)	25	0031	0x19	9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
(sub)	26	0032	0x1a	:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
(esc)	27	0033	0x1b	;	59	0073	0x3b	[	91	0133	0x5b	{	123	0173	0x7b
(fs)	28	0034	0x1c	<	60	0074	0x3c	\	92	0134	0x5c	|	124	0174	0x7c
(gs)	29	0035	0x1d	=	61	0075	0x3d	]	93	0135	0x5d	}	125	0175	0x7d
(rs)	30	0036	0x1e	>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
(us)	31	0037	0x1f	?	63	0077	0x3f	_	95	0137	0x5f	(del)	127	0177	0x7f

字串(字元陣列)

- C的字串由字元陣列構成
  - 例如: char c[10];
- 字串結尾時使用 0 (NULL) 結束
- 範例程式:

```
char c[4];
c[0]='H';
c[1]=105;
c[2]=0;
printf("%s\n",c);
```

c[0]	c[1]	c[2]	c[3]
H	i	0(Null)	?

有用的字串處理函式簡介

標題檔: <string.h>

常用函式:

strcpy(char* *D*,char* *S*); 將字串S拷貝到D

strcat(char **D*, char* *S*); 將字串S接到到D的後面

int strcmp(char *D,char* S); 比較字串S和D,傳回值爲0時代表相同

範例程式:

```
char D[20], E[20];
 strcpy(D,c);
 strcat(D,"story");
 printf("%s\n",D);
 while(strcmp(D,E)!=0)
 {
  printf("Input a string:");
  scanf("%s",E);
 }
```

課堂練習

- 計算自己ASCII碼的生命數字
- For example:

B	I	L	L	G	A	T	E	S	I	I	I
66	73	76	76	71	65	84	69	83	1	1	1

- 總合=666,惡魔的象徵
  - 總合不要用char宣告,會造成溢位(i.e.:>255)
- 相關網頁

檔案的處理

- 在C語言中,每個開啓的檔案都用一個代號(file handle)來指定.
- 檔案處理流程
  - 檔案開啓: FILE* fopen(“filename”, “attribute”)
    - Attribute: “r”=>唯讀, “w”=>寫入 “a”=>附加在檔案後
  - 檔案的存取:
    - 檔案的讀取: fscanf(FILE*, “%f %c …”, variable names)
    - 寫入檔案: fprintf(FILE*, “%f %c …”, variable names)
  - 檔案的關閉: fclose (FILE*)
    - 檔案不關閉可能會造成資料遺失

範例程式

```
FILE* Read=fopen("ReadTest.txt","r");
        開啓檔案ReadText.txt,僅供讀取
FILE* Write=fopen("WriteTest.txt","w");
              開啓檔案WriteText.txt,以供輸出
char Name[4][20], Date[20];
float Index[3];
 fscanf(Read,"%s %s %s %s", Name[0],Name[1],Name[2],Name[3]);
          讀取ReadText.txt的第一行
printf("%7s %7s %7s %9s\n", Name[0],Name[1],Name[2],Name[3]);
          輸出ReadText.txt的第一行(用%7s決定輸出字串長度)
while(fscanf(Read,"%s %f %f %f",Date,&Index[0],&Index[1],&Index[2])!=EOF)
{             讀取資料,直到檔案結束(EOF)
  printf("%7s %7.2f %7.2f %7.2f\n",Date,Index[0],Index[1],Index[2]);
              用%7s, %7.2f 決定輸出字串長度)
  fprintf(Write, "%f %f %f %s\n", Index[0],Index[1],Index[2],Date);
               將資料寫入 WriteTest.txt
}

fclose(Write);
fclose(Read);  檔案關閉
```

比較ReadTest 和WriteTest不同

電腦用二進位儲存資料,所以在進位制互換時,會有些微的誤差

ReadTest.txt

Date	台指	電子指	金融指
2003/6/2	4692.94	198.79	692.21
2003/6/3	4678.08	198.47	698.83
2003/6/5	4738.34	201.08	695.00
2003/6/6	4740.45	202.27	706.84

WriteTest.txt

```
4692.939941 198.789993 692.210022 2003/6/2
4678.080078 198.470001 698.830017 2003/6/3
4738.339844 201.080002 695.000000 2003/6/5
4740.450195 202.270004 706.840027 2003/6/6
```

課堂演練

- 以寫入模式開啓檔案” Average.txt”,
  - fopen(“Average.txt”,”w”);
- 更改上述程式,分別計算台指,電子指,和金融指在這幾天的指數平均值
- 將計算的結果放入” Average.txt”中
- 記得關閉檔案
  - fclose();

日期的計算

- 處理財務程式時,常碰到天數計算的問題
- 處理天數計算的程式 (請見Calendar project mon.cpp)
  - calendar.cpp 處理曆法的變革→for history study
- 程式中輸入起訖的年月日,即可算出這段時間相隔幾天

  count_t(int year1,int month1, int day1,int year2,int month2, int day2)
- 比較Excel檔中的日期格式
  - 2003/6/2
    - 都用 “/”區隔
  - 格式轉換從字串 “2003/6/2”換成→ 2003 6 2 (三個integers)

日期的計算

- c

2	0	0	3	/	6	/	2	Null

- Year

2	0	0	3	Null

Y=atoi(Year) (included in stdlib.h)

- 用integer 表年份

程式碼

```
char c[15];
char Year[5]; int Y;
cin>>c;
int p0=0,p1=0;
while(c[p0]!='/')
{
 Year[p1]=c[p0];
 p0=p0+1;
 p1=p1+1;
}
Year[p1]=0;
Y=atoi(Year);
cout<<Y;
```

課堂練習

- 輸入兩個字串
  - Ex: “2003/1/1” ….
  - 放入Start[]和End[]中
- 利用上式轉換成用整數表示的年月日
  - Start[]已放入→ year1,month1,day1
  - 寫程式將End[]放入→year2,month2,day2
- 呼叫天數計算程式計算其相隔的天數

  count_t(int year1,int month1, int day1,int year2,int month2, int day2)

使用文字檔和Excel交換資料

- Excel提供
  - 讀取文字檔
  - 將工作表的內容儲存成文字檔
- 藉此提供程式和excel之間資料轉換的橋樑
- Excel檔轉存文字檔
  - 使用 Granvile project中的 Read.xls
  - 另存新檔 (R.txt)
  - 將檔案儲存格式設爲文字檔

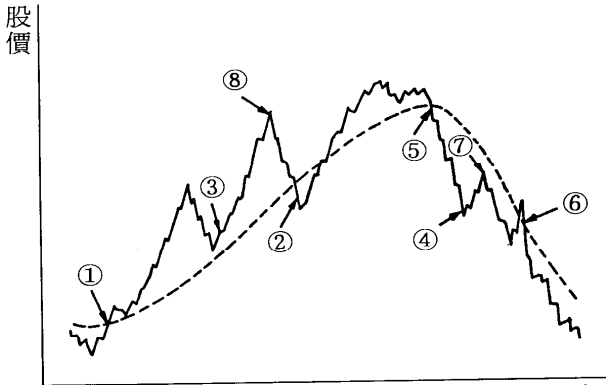
檢驗葛蘭碧移動平均線法則

- 移動平均計算標的物在某一段期間內的平均價格
- 以三日平均線爲例:

時間	t	t+1	t+2	t+3	t+4
標的物價格	30	29	31	33	32
移動平均			30	31	32

- 葛蘭碧分析移動平均線和股價指數的關係,建立了葛蘭碧移動平均線八法則
  - 撰寫程式用歷史資料檢驗其中兩個法則的正確性

檢驗葛蘭碧移動平均線法則

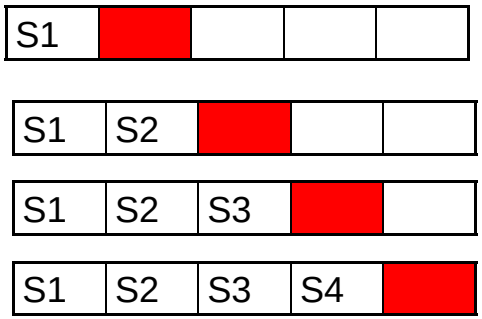


取週線(5交易日)當作判斷的移動平均線,使用左述兩個法則決定買進和賣出的時間點,並用實際的歷史資料檢驗

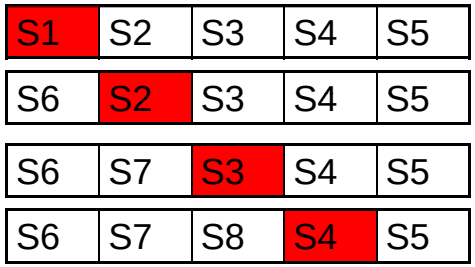
1.平均線上升,股價向上超越平均線爲買進訊號
5.平均線下降,股價向下超越平均線爲賣出訊號

檢驗葛蘭碧移動平均線法則
(使用陣列處理移動平均)

- 使用環狀陣列的概念,假想陣列頭尾相連
- 假定標的物的價格=>S1,S2,S3,…
- 則五日平均可計算如下:



檢驗葛蘭碧移動平均線法則
(使用陣列處理移動平均)

- 

對應程式碼:

```
while(fscanf(Read,"%s %f %f %f",Date,&Index[0],&Index[1],&Index[2])!=EOF)
 {
   MAArray[ArrayPtr]=Index[0];
   ArrayPtr=(ArrayPtr+1)%5;
   for(int i=0;i<5;i++)            平均值的計算
   { Avg=Avg+MAArray[i];}
   Avg=Avg/5;
 }
```

檢驗葛蘭碧移動平均線法則
(買進和賣出法則的判斷)

- 變數名稱及其涵義
  - Buy=>持有部位=1,未持有部位=0
  - Avg=>現在移動平均, PreAvg=>前一日移動平均
  - Profit=>損益, BuyCost=>購買成本
- 符合法則一時買入

```
if((Buy==0)&&(Avg>=PreAvg)&&(Index[0]>=Avg))
{//未持有部位,移動平均大於前一日移動平均(移動平均上升),股價超越平均
 BuyCost=Index[0];
 Buy=1;}
```

- 符合法則五時賣出

```
if((Buy==1)&&(Avg<=PreAvg)&&(Index[0]<=Avg))
{//持有部位,移動平均小於前一日移動平均(移動平均下降),股價向下穿過平均
 Profit=Profit+(Index[0]-BuyCost);
 Buy=0;}
```

用while迴圈讀取資料

```
while(fscanf(Read,"%s %f %f %f",Date,&Index[0],&Index[1],&Index[2])!=EOF)
 {
   MAArray[ArrayPtr]=Index[0];
   ArrayPtr=(ArrayPtr+1)%5;
   Count=Count+1;
   if(Count<5) continue;                                  ← 五筆資料以上才處理
            PreAvg=Avg;
            Avg=0;
            for(int i=0;i<5;i++)
            { Avg=Avg+MAArray[i];}
            Avg=Avg/5;
   if(Count==5)  continue;
        if((Buy==0)&&(Avg>=PreAvg)&&(Index[0]>=Avg))      ← 判斷買進賣出
        {//買進
         BuyCost=Index[0];
         Buy=1;
        }
        if((Buy==1)&&(Avg<=PreAvg)&&(Index[0]<=Avg))
        {//賣出
         Profit=Profit+(Index[0]-BuyCost);
         Buy=0;
        }
        fprintf(Write, "%s %f %f %f %d\n", Date, Index[0], Avg, Profit, Buy);
 }
  if(Buy==1)
  {                                                       最後一日強迫平倉
          Profit=Profit+(Index[0]-BuyCost);
  }
fprintf(Write,"平倉: %f",Profit);
```

課堂演練
葛蘭碧移動平均線法則

- 修改上述程式,計算並輸出電子指的移動平均
  - 移動平均改採半月線(10日)
- 仿照上述程式,以電子指當標的物,驗證葛蘭碧移動平均線法則一和五的正確性

對數常態分配下價格機率模型估計

- 假定資產價格的隨機過程服從對數常態分配

$$\ln \frac{s_i(j+1)}{s_i(j)} \sim N\left(\left(\mu_i - 0.5\sigma_i^2\right)\tau, \sigma_i^2\tau\right)$$

- n個資產：S_1、S_2 … S_n；$S_i(j)$代表第j個時間S_i的價格(觀察m+1次，間隔時間 τ)
- 資產S_i的隨機過程可表示：

$$S_i(j) = S_i(0)e^{(\mu_i - 0.5\sigma_i^2)j\tau + \sigma_i B_i(j\tau)}$$

- 但如何估計 μ_i, σ_i？

- 令
  - $r_i(j) = \ln(\frac{S_i(j+1)}{S_i(j)})$ ， $\bar{r}_i = \frac{\sum_{j=1}^{m} r_i(j)}{m}$
  - 則S_i的平均報酬率 μ_i 可估計爲 $\frac{\bar{r}_i}{\tau}$
- 令s_i爲$r_i(j)$的標準差的估計量
  - $s_i^2 = \frac{\sum_{j=1}^{m}(r_i(j)-\bar{r}_i)^2}{m-1} = \frac{\sum_{j=1}^{m}(r_i(j))^2}{m-1} - \frac{\left[\sum_{j=1}^{m} r_i(j)\right]^2}{m(m-1)}$
  - 則 σ_i^2 的估計量爲 $\frac{s_i^2}{\tau}$

不同資產價格變動之關連性

- S_i和S_j報酬率的共變異數 σ_{jk}^2 可估計如下

$$\frac{\sum_{j=1}^{m}[(r_i(j)-\bar{r}_i)(r_k(j)-\bar{r}_k)]}{(m-1)\tau} = \frac{\sum_{j=1}^{m}(r_i(j)r_k(j))}{(m-1)\tau} - \frac{\left(\sum_{j=1}^{m} r_i(j)\right)\left(\sum_{j=1}^{m} r_k(j)\right)}{m(m-1)\tau}$$

- n 個資產的平均報酬率向量 μ

$$\mu = (\mu_1, \mu_1, \cdots \mu_n)$$

- 共變異數矩陣 $\sum = \begin{bmatrix} \sigma_1^2 & \sigma_{12}^2 & \cdots & \sigma_{1n}^2 \\ \sigma_{21}^2 & \sigma_2^2 & \cdots & \sigma_{2n}^2 \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_{n1}^2 & \sigma_{n2}^2 & \cdots & \sigma_n^2 \end{bmatrix}$

程式範例：估計台指和電子指的 μ 和 Σ

Step 1. 開啓資料檔案和變數宣告。

Step 2. 讀取資料並計算 $\sum_{j=1}^{m} r_k(j)$ (指數報酬率的和)、$\sum_{j=1}^{m}(r_i(j))^2$ (平方和)以及 $\left(\sum_{j=1}^{m} r_i(j)r_k(j)\right)$ (台指及電子指報酬率乘績的和)。

Step 3. 利用Step2所求之計算台指和電子指報酬率的期望值、變異數和共變異數。

Step 4. 輸出計算結果。

Step2 讀取資料 & 計算

請參見Estimate project

用while迴圈讀取資料

```
int m=0,First=1;
while(fscanf(Read,"%s %f %f %f",Date,&Index[0],&Index[1],&Index[2])!=EOF)
{
 if(First)
        {
         S1=Index[0];
         S2=Index[1];
         First=0;
         continue;
        }
     r1=log(Index[0]/S1);
     r2=log(Index[1]/S2);
     r1Sum=r1Sum+r1;
     r1Square=r1Square+r1*r1;
     r2Sum=r2Sum+r2;
     r2Square=r2Square+r2*r2;
     r1r2Sum=r1r2Sum+r1*r2;
     S1=Index[0];
     S2=Index[1];
     m=m+1;
}
```

第一筆資料先不處理

計算

$\sum_{j=1}^{m} r_i(j)$

$\sum_{j=1}^{m}(r_i(j))^2$

$\left(\sum_{j=1}^{m} r_i(j)r_k(j)\right)$

Step3 計算報酬率期望值、變異數和共變數

```
Mu1=r1Sum/(m*Tau);
Mu2=r2Sum/(m*Tau);
Sigma1Square=(r1Square/(m-1)-r1Sum*r1Sum/(m*(m-1)))/Tau;
Sigma2Square=(r2Square/(m-1)-r2Sum*r2Sum/(m*(m-1)))/Tau;
Sigma12=r1r2Sum/((m-1)*Tau)-r1Sum*r2Sum/(m*(m-1)*Tau);
```

Step4 輸出計算結果

```
printf("平均報酬率向量= (%lf,%lf)\n",Mu1,Mu2);
printf("共變異數矩陣:\n%lf\t%lf\n%lf\t%lf\n",
        Sigma1Square,Sigma12,Sigma12,Sigma2Square);
```

執行結果

```
平均報酬率向量= (0.599021,0.647597)
共變異數矩陣：
0.032636        0.038141
0.038141        0.051033
```

課堂演練

- 讀入台指期、電子期、金融期的每日歷史資料，並計算這三種期貨指數的平均報酬率　　和報酬率的共變異矩陣