

004-CE

電腦科

第一
乙部及
丙部)

香港考試及評核局

2004年香港中學會考

電腦科 試卷一（乙部及丙部） 試題答題簿

本試卷必須用中文作答

兩小時完卷(上午八時三十分至上午十時三十分)

考生須知：

1. 在指定位置填寫考生編號、試場編號及座位編號。
2. 本卷試題全部均須作答。
3. 考生須在本卷指定位置上作答。
4. 如有需要，可要求派發補充答題紙。每張紙均須寫上考生編號，並用繩縛於簿內。

考生編號									
試場編號									
座位編號									

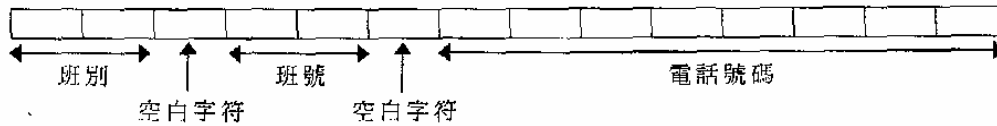
由閱卷員填寫	由試卷主席填寫
閱卷員編號	試卷主席編號

Q.6										
Q.7										
Q.8										
Q.9										
Q.10										
Q.10	0	1	2	3	4	0	1	2	3	4
總分										

核分員專用	
核分員編號	
總分	

乙部 (40 分)
本部各題均須作答。

6. 某學校將學生的電話號碼存貯在一個文字檔 'telist.txt'。在這文字檔中，每行有 14 個字符，代表學生班別、班號和電話號碼，由空白字符分隔，如下所示：



部分文字檔如下所示：

```
1B 12 23516718
1C 23 24567171
1C 12 25167717
1D 04 24516818
1E 23 21346765
.
.
```

陳先生編寫了一個程序來改動學生的電話號碼。這個程序需把該文字檔更新，並在 VDU 上顯示如下的輸出。（在此輸出中，各冒號後面的數據均由用戶以鍵盤輸入，其餘則由程序輸出。）

樣本輸出 1

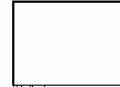
```
Enter class name : 1D
Enter class no.: 04
Enter new telephone no.: 26743219
Record has been modified successfully.
```

樣本輸出 2

```
Enter class name : 1E
Enter class no.: 16
No such record!
```

陳先生的程序如下所示：

行號	程序語句
100	Program Q6;
110	var datafile : text;
120	data: array [1..200] of string;
130	inpno, inpclass, oneline, newno :string;
140	class, classno: string[2];
150	position, i, count :integer;
160	begin
170	assign(datafile, 'telist.txt');
180	reset(datafile);
190	write('Enter class name : ');
200	readln(inpclass);
210	write('Enter class no.: ');
220	readln(inpno);



行號 程序語句

```
230      count:=0;
240      position:=0;
250      while not eof(datafile) do
260          begin
270              count:=count + 1;
280              readln(datafile, oneline);
290              data[count]:=oneline;
300              class:=copy(data[count], 1, 2);
310              classno:=copy(data[count], 4, 5);
320              if (class=inpclass) or (classno=inpno) then
330                  position:=count
340          end;
350      if position = 0 then
360          begin
370              write('Enter new telephone no.: ');
380              readln(newno);
390              data[position]:=newno;
400              reopen(datafile);
410              for i := 0 to count do
420                  writeln(datafile, data[count]);
430          end;
440      writeln('No such record!');
450      writeln('Record has been modified successfully. ');
460      close(datafile);
470      end.
```

然而，在程序中行號第 160 行以後有多個錯誤。填妥下表，以示所有錯誤的位置及更正後的語句。

行號	更正後的語句

(10 分)

7. 小明編寫了一個程序來把一段文字加密。這個程序只處理大楷字母，如下所示：

```
program Code;
const shift = 13;
var
    InStr  : string;
    i, tmp : integer;
begin
    write('Please input the text : ');
    readln(InStr);
    write('The coded message is : ');
    for i := 1 to length(InStr) do
        if (InStr[i] >= 'A') and (InStr[i] <= 'Z') then
            begin
                tmp := ord(InStr[i]) + shift;
                if tmp > 90 then tmp := tmp - 26;
                write(chr(tmp))
            end
        else
            write(InStr[i]);
        writeln
    end.
end.
```

- (a) 當輸入下列文字後，試寫出該程序的輸出。

(i) BAR

(ii) ONE

(2 分)

- (b) 試簡述如何利用上列程序把加密文字解碼。

(1 分)

小芬獲得小明的加密文字。她被告知可利用下列步驟，破譯小明的加密文字：

步驟 1：數出在加密文字中每個字母出現的次數。

步驟 2：一般來說，從統計數字得知 'E' 是在文字段落中最常用的字母。因此，小芬將在加密文字中出現次數最多的字母更換為 'E'，把下一個按字母順序排列的字母更換為 'F'，如此類推。例如，若 'Y' 是在加密文字中出現次數最多的字母，便將 'Y' 更換為 'E'，其後更換 'Z' 為 'F'，更換 'A' 為 'G'，更換 'B' 為 'H'，.....，如此類推。



- (c) 小芬編寫了一個程序，數出用戶指定文字檔內，每個字母出現的次數，並找出出現次數最多的字母。試寫出下列方格內的內容來完成這個程序。

```
program Count;
var
  filename, InStr : string;
  Infile :  ;
  Freq : array['A'..'Z'] of integer;
  c, max : char;
  i : integer;
begin
  for c := 'A' to 'Z' do
    Freq[c] := 0;
  write('Please input the name of the file to be processed : ');
  readln(filename);
  assign(Infile,  );
  reset(Infile);
  while not eof(Infile) do
    begin
      readln(infile, InStr);
      for i := 1 to  do
        if (InStr[i] >= 'A') and (InStr[i] <= 'Z') then
          Freq[InStr[i]] :=  ;
      end;
      close(Infile);
      max := 'A';
      i := Freq[max];
      for c := 'B' to 'Z' do
        if  then max := c;
      writeln('The most frequent letter is ', max);
    end.
```

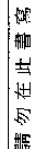
_____ _____ _____
 _____ _____

(5 分)

- (d) 試舉出這個方法，破譯只含大楷字母段落的兩個弱點。

(2 分)

請勿在此處寫字



8. 下列算法計算此表達式： $\frac{1}{2}(f_1 + 2f_2 + 2f_3 + \dots + 2f_{n-1} + f_n)$

步驟 1: 說明 i 和 n 為整數變量。

說明 f 和 result 爲實數變量。

步驟 2： 初始化 i 為 1。

初始化 result 為 0。

步驟 3：讀入一個整數至變量 n 。

步驟 4: 常變量 i 的值少於或等於變量 n 的值, 執行步驟 5 至 7。

步驟 5: 讀入一個實數至變量 f 。

步驟 6： 若變量 i 的值是不等於 1 和不等於變量 n 的值，
將變量 result 的值加上變量 f 兩倍的值，
否則將變量 result 的值加上變量 f 的值。

步驟 7: 將變量 i 的值加 1。

步驟 8： 將變量 result 的值除以 2。

步驟 9：顯示變量 result 的值。

試將此算法改寫成一個 Pascal 程序。 n 和 f_i ($i = 1, 2, 3, \dots, n$) 的值在執行這個程序時，透過鍵盤順序輸入。

[illegible]

(10 分)

9. 剔除了空白，正讀和反讀都一樣的詞或句子，稱為迴文 (palindrome)。兩個迴文的例子如下所示：

- (1) Rats live on no evil star
- (2) Was it a car or a cat I saw

彼得編寫了一個程序，查證一段字符串是否迴文。下列為程序的樣本輸出：（在此輸出中，各冒號後面的數據均由用戶以鍵盤輸入，其餘則由程序輸出。）

樣本輸出 1

```
Please input a string: Was it a car or a cat I saw
It is a palindrome.
```

樣本輸出 2

```
Please input a string: no pain no gain
It is not a palindrome.
```

彼得已編寫好程序的說明及主體部分，但有些過程還未完成，如下所示：

```
program Palindrome;
var
  inp, nospace : string;
  i, j : integer;
  result : boolean;
```

未完成的過程

```
begin
  write('Please input a string: ');
  readln(inp);
  RemoveSp;
  Upper;
  result := true;
  Test(result);
  if result then
    writeln('It is a palindrome.')
  else writeln('It is not a palindrome.');
```

end.

The diagram shows a rectangular domain with a central square hole. The domain is divided into four quadrants by a vertical line at $x=0$ and a horizontal line at $y=0$. The central square hole is centered at the origin. The domain is labeled with x and y axes. The central square hole is labeled 'hole'.

標識符	描述
Inp	字符串變量，用來存貯輸入字符串
nospace	字符串變量，用來存貯剔除了空白字符的字符串
i, j	整數變量，用作計數器
result	布爾變量，用來存貯查證的結果

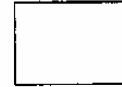
作答本題時，考生不得加入新的變量，否則不獲給分。

- (a) 編寫過程 RemoveSp，剔除存貯在變量 inp 內的字符串中的空白字符，並把這字符串存貯至變量 nospace。

```
procedure RemoveSp;
begin
```

end;

(3 分)



- (b) 編寫過程 Upper，將變量 nospac 中的字母改成大楷。

```
procedure Upper;  
begin
```

```
_____  
_____  
_____  
_____  
_____  
_____  
end;
```

(3 分)

- (c) 編寫過程 Test，查證變量 nospac 中的字符串是否迴文。

```
procedure Test(var x : boolean);  
begin
```

```
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
end;
```

(4 分)

- 乙部完 -



丙部 (本部佔 16 分。考生如能在本題展示有效的傳意技巧，可另得分，最高可獲 4 分。)

10. 李先生是 ABC 大學註冊部主管，他欲將課程註冊系統電腦化。他設計了兩個數據庫檔：Personal 和 CourseReg，分別用作存貯學生的個人資料和學生選修的科目。

Personal

欄名	類型	備註
Name	字符	學生姓名
SID	字符	學生編號
Sex	字符	性別
DOB	字符	出生日期
TelNo	字符	電話號碼
Addr	字符	居住地址
Email	字符	電郵地址

CourseReg

欄名	類型	備註
SID	字符	學生編號
SubCode1	字符	第一科的科目編碼
SubCode2	字符	第二科的科目編碼
SubCode3	字符	第三科的科目編碼
SubCode4	字符	第四科的科目編碼
SubCode5	字符	第五科的科目編碼

以下是 CourseReg 的一個例子：

SID	SubCode1	SubCode2	SubCode3	SubCode4	SubCode5
127768	012	124	025		
453860	105	124	025	023	028
049532	012	110	011	039	041
454890	025	033	011	023	
.	.	.	.	.	.
.	.	.	.	.	.

當選修科目的數目少於五，有些科目編碼的欄將會漏空。

- (a) (i) 哪個欄該能獨一無二地確定一個記錄？

(ii) 為什麼 SID 的數據類型是字符而非數字？



(iii) 在 CourseReg 裏包括了科目編碼而非科目名稱。試寫出兩個優點。

(4 分)

李先生修改了 CourseReg 的數據庫結構，如下所示：

CourseReg

欄名	類型	備註
SID	字符	學生編號
SubCode	字符	科目編碼

以下是 CourseReg 的一個例子：

SID	SubCode
127768	012
127768	124
127768	025
453860	105
453860	124
453860	025
453860	023
453860	028
.	.
.	.
.	.

(b) 試描述這次修改如何對此系統有利。

(2 分)

請勿在此
作答



ABC 大學 (註冊紙)

姓名：

學生編號：

性別：

電郵地址：

科目編碼 1：

科目編碼 2：

科目編碼 3：

科目編碼 4：

科目編碼 5：

在註冊的時候，每名學生需填寫一張註冊紙，如上圖所示，並交回註冊處。小強是該大學的一位文員，他會把這些數據輸入電腦。李先生需要在數據輸入進程中避免所有出錯。因此，小強計劃將這些數據鍵入兩次，但是李先生認為這樣不是一個好方法。

(c) (i) 試舉出小強的計劃兩個缺點。

(ii) 爲了改善數據輸入進程，李先生欲

- (1) 重新設計註冊紙讓學生填寫，
- (2) 將輸入進程自動化。

試說明怎樣把這兩件事做好。

(6 分)

請在此處寫備



- (d) 現在這個輸入進程應該是全無錯誤的，但是電腦仍然會發出與數據有關的錯誤報告。試舉出**兩個**可能出現的錯處。

(2 分)

- (e) 學生需要校驗他們在該系統內的個人資料和課程註冊詳情。小強計劃把所有學生資料，包括 Personal 和 CourseReg 的所有內容，清楚貼於註冊處外的佈告板上，讓學生查證自己的資料。

李先生並不同意小強的做法。為什麼？小強應該怎樣處理？

(2 分)

- 丙部完 -

- 試卷完 -

請勿在此處寫

由閱卷員填寫	
R	
L	

美國信息交換標準代碼 (ASCII) 編碼表

字符	ASCII	字符	ASCII	字符	ASCII
0	48	J	74	d	100
1	49	K	75	e	101
2	50	L	76	f	102
3	51	M	77	g	103
4	52	N	78	h	104
5	53	O	79	i	105
6	54	P	80	j	106
7	55	Q	81	k	107
8	56	R	82	l	108
9	57	S	83	m	109
:	58	T	84	n	110
;	59	U	85	o	111
<	60	V	86	p	112
=	61	W	87	q	113
>	62	X	88	r	114
?	63	Y	89	s	115
@	64	Z	90	t	116
A	65	[	91	u	117
B	66	\	92	v	118
C	67	]	93	w	119
D	68	^	94	x	120
E	69	_	95	y	121
F	70	`	96	z	122
G	71	a	97	{	123
H	72	b	98		124
I	73	c	99	}	125

可用運算符及保留字表

!, -, *, /, div, mod, >, <, >=, <=, <>, and, or, not, sqrt, trunc, round, random, abs, +(字符串連接), length, ord, chr, val, str, copy, const, procedure, var, integer, real, char, string, boolean, true, false, text, array...of, program, input, output, begin...end, :=, for...to/downto...do, (*...*), if...then...else, while...do, repeat...until, case...of...end, write, writeln, read, readln, assign, reset, rewrite, eof, close