

	Format 類型	# Questions 題目數	Total Marks 佔分
Section A1 甲部 (一)	True or False 真假題	5	5
Section A2 甲部 (二)	Multiple Choice 多項選擇題	20	20
Section B 乙部	Fill-in-the-blanks 填充題	7 (A - M)	20
Total 總分			45

- (1) Assume that all variables without declaration shown in the following program segments have already been declared properly as 32-bit signed integers (Pascal: `longint`, C / C++: `int`).
下列程序段中所有未有列出宣告的變量，均假設已經適當地宣告為 32 位元有符號的整數 (Pascal: `longint`, C / C++: `int`)。
- (2) The following code is added to the beginning of all C and C++ programs.
在所有 C 和 C++ 程序的頂部加入以下程式碼:

C

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
#include <stdbool.h>
```

C++

```
#include <cstdio>
#include <cmath>
#include <cstdlib>
#include <string>
#include <iostream>
using namespace std;
```

For C, `stdbool.h` defines the boolean data type `bool` and values `true` (equivalent to 1) and `false` (equivalent to 0).

對於 C，`stdbool.h` 定義了布爾數據類型 `bool` 及值 `true` (等同 1) 及 `false` (等同 0)。

- (3) Assume all programs are compiled properly in Ubuntu 16.04 using the compilers and commands below.
假設所有程序都在 Ubuntu 16.04 下使用以下編譯器及指令正確地編譯。

```
Pascal: Free Pascal (fp-compiler 3.0.0)   fpc program.pas
C:      GNU GCC (gcc-4.9 4.9.3)         gcc -std=c99 program.c -o program
C++:   GNU G++ (g++-4.9 4.9.3)       g++ -std=c++98 program.cpp -o program
```

Section A1 甲部 (一) (5 marks 分)

For each question, determine whether the statement is true or false, then mark the corresponding box (T: true or F: false) on the answer sheet. One mark for each correct answer. No marks will be deducted for wrong answers.
請判斷下列每題的陳述句的真假，然後把答題紙對應的空格 (T: 真或 F: 假) 填滿。答對得一分，答錯不扣分。

1. Users can improve their computers' performance by downloading more Random-Access Memory (RAM) from the Internet.
用戶可從互聯網下載更多隨機存取記憶體 (RAM) 以改善電腦的效能。
2. USB 2.0 has a lower data transfer rate than USB 3.0 because USB 2.0 data signals are analog.
USB 2.0 的傳輸速度較 USB 3.0 低是因為 USB 2.0 資料信號是模擬的。
3. The size of a character (char) variable is 1 byte.
一個字符 (char) 變量的大小是一個位元組 (字節)。
4. Suppose x is a boolean (Pascal: `boolean`, C/C++: `bool`) variable. Evaluate the following expression.
若 x 是一個布爾 (Pascal: `boolean`, C/C++: `bool`) 變量，那麼以下表達式的運算結果是什麼？

Pascal

`(x and (not x)) = false`

C / C++

`(x && (!x)) == false`

5. There exists an equilateral triangle (with positive area) on the Cartesian coordinate plane, such that the coordinates of its three vertices are all integers.
笛卡兒座標平面上存在一個三點座標均為整數 (且面積為正) 的等邊三角形。

Section A2 甲部 (二) (20 marks 分)

For each question, choose the **most appropriate** answer and mark the corresponding box (A, B, C, or D) on the answer sheet. One mark for each correct answer. No marks will be deducted for wrong answers.

請為下列每題各選一個**最適合**的答案，然後把答題紙對應的空格 (A、B、C、或 D) 填滿。答對得一分，答錯不扣分。

6. Ransomware is a class of computer virus that usually affects users' computers by:
勒索軟體一種電腦病毒，它通常以以下哪種方式影響電腦用家?
 - A. Turning off the computer suddenly
突然把電腦關機
 - B. Displaying advertisements repeatedly
不斷展示廣告
 - C. Making annoying sounds
發出煩擾聲音
 - D. Encrypting files
把檔案加密

7. Which of the following data structure is most suitable for simulating a recursion?

以下哪一項數據結構最適合用來模擬遞歸？

- A. Stack 堆疊 (棧)
- B. Queue 隊列
- C. Linked list 鏈表
- D. Hash table 哈希表

8. Assume function `random(n)` returns an integer from 0 to `n-1` randomly with equal probability. Consider the following program:

假設函數 `random(n)` 會以相等的概率隨機地傳回一個由 0 至 `n-1` 之間的整數。考慮以下程序：

Pascal

```
var i, a, b: longint;  
begin  
  a := random(4) * 2;  
  b := random(3) + random(3);  
  write(b - a)  
end.
```

C

```
int i, a, b;  
int main() {  
  a = random(4) * 2;  
  b = random(3) + random(3);  
  printf("%d", b - a);  
  return 0;  
}
```

C++

```
int i, a, b;  
int main() {  
  a = random(4) * 2;  
  b = random(3) + random(3);  
  cout << b - a;  
  return 0;  
}
```

What is the number of possible different outputs? 有多少種輸出的可能性？

- A. 10
 - B. 11
 - C. 12
 - D. 13
9. Given that there is only one correct answer for this question, which one is the correct answer?

如果這條題目只有一個正確答案的話，那麼這個正確的答案是？

- A. Neither B nor C is the correct answer. B 和 C 都不是正確答案。
- B. Neither C nor D is the correct answer. C 和 D 都不是正確答案。
- C. Trust me, this is the correct answer. 相信我，這是正確答案。
- D. Neither A nor B is the correct answer. A 和 B 都不是正確答案。

10. Go is a board game in which the two players, black and white, take turns to put a stone on a 19×19 board. The goal is to surround more territory than the opponent. Which of the following plays Go best?

圍棋 (Go) 是一種雙人棋類遊戲。對弈雙方在一個 19×19 的棋盤網格上交替放置黑色和白色的棋子，擴大己方棋子所圍的區域。以下哪一項最擅長下圍棋？

- A. Deep Blue 深藍
- B. Pokémon GO
- C. AlphaGo
- D. DuckDuckGo

11. Which of the following program segments is an implementation of bubble sort?

n is the size of the array `int a[n]` (Pascal: `a: array[0..n-1]` of `longint`).

以下那一段程序段是冒泡排序法的實現？

n 是陣列 `int a[n]` (Pascal: `a: array[0..n-1]` of `longint`) 的大小。

Pascal

C / C++

A.

```
for i := 0 to n - 1 do
  for j := i downto 1 do
    if (a[j] < a[j - 1]) then
      begin
        temp := a[j];
        a[j] := a[j - 1];
        a[j - 1] := temp
      end;
```

```
for (i = 0; i <= n - 1; i++)
  for (j = i; j >= 1; j--)
    if (a[j] < a[j - 1]) {
      temp = a[j];
      a[j] = a[j - 1];
      a[j - 1] = temp;
    }
```

B.

```
for i := 0 to n - 1 do
  for j := i to n - 2 do
    if (a[j] > a[j + 1]) then
      begin
        temp := a[j];
        a[j] := a[j + 1];
        a[j + 1] := temp
      end;
```

```
for (i = 0; i <= n - 1; i++)
  for (j = i; j <= n - 2; j++)
    if (a[j] > a[j + 1]) {
      temp = a[j];
      a[j] = a[j + 1];
      a[j + 1] = temp;
    }
```

C.

```
for i := 0 to n - 1 do
  for j := 0 to i - 1 do
    if (a[j] > a[j + 1]) then
      begin
        temp := a[j];
        a[j] := a[j + 1];
        a[j + 1] := temp
      end;
```

```
for (i = 0; i <= n - 1; i++)
  for (j = 0; j <= i - 1; j++)
    if (a[j] > a[j + 1]) {
      temp = a[j];
      a[j] = a[j + 1];
      a[j + 1] = temp;
    }
```

D.

```
for i := 0 to n - 1 do
  for j := n - 1 downto i + 1 do
    if (a[j] < a[j - 1]) then
      begin
        temp := a[j];
        a[j] := a[j - 1];
        a[j - 1] := temp
      end;
```

```
for (i = 0; i <= n - 1; i++)
  for (j = n - 1; j >= i + 1; j--)
    if (a[j] < a[j - 1]) {
      temp = a[j];
      a[j] = a[j - 1];
      a[j - 1] = temp;
    }
```

12. How many ways are there to pick, at the same time, two different integers between 1 and 8, such that their product is **not** a multiple of four?

有多少種不同的方法在 1 和 8 之間同時選兩個不同的整數，使得它們的乘積**不是** 4 的倍數？

- A. 13
- B. 14
- C. 15
- D. 16

13. Alice and Bob are playing a game on a $N \times M$ chessboard. Initially, the chessboard is empty. The players take turn to place a rook on the chessboard, with the restriction that the rook cannot attack any other rooks on the chessboard. The player who first runs out of moves loses.

One rook can attack another piece if and only if:

1. they are on the same row or on the same column, and
2. there are no other pieces between them.

愛麗絲和鮑伯在一個 $N \times M$ 的棋盤上下棋。初始時，棋盤上沒有任何棋子。他們輪流在棋盤上放一隻城堡，而該隻城堡不能攻擊其他城堡。不能放棋子的玩家為輸。

一隻城堡能攻擊另一隻棋子當且僅當:

1. 它們位於相同的行或列，以及
2. 它們之間沒有其他棋子。

Alice goes first. For what board sizes (N, M) does Alice have a winning strategy?

如果愛麗絲是先手，她在以下哪一組 (N, M) 的棋盤有必勝的策略？

- i. $(N, M) = (3, 5)$
- ii. $(N, M) = (8, 8)$
- iii. $(N, M) = (7, 6)$

- A. i only 只有 i
B. ii only 只有 ii
C. i and iii only 只有 i 和 iii
D. ii and iii only 只有 ii 和 iii

14. Consider the following function: 考慮以下函數:

Pascal

```
function f(x: longint): longint;  
begin  
  if x = 2 then  
    f := x - 2  
  else  
    f := x * 2  
end;
```

C / C++

```
int f(int x) {  
  if (x == 2)  
    return x - 2;  
  else  
    return x * 2;  
}
```

Here, x is an integer in the range $[0, 2^{31} - 1]$. Which of the following is/are true?

以上函數中的 x 是一個範圍是 $[0, 2^{31} - 1]$ 的整數。以下哪些為真？

- i. $f(x)$ must be an even number. $f(x)$ 必定是雙數。
 - ii. $f(x)$ must be a non-negative number. $f(x)$ 必定是非負數。
- A. None of the above 以上皆否
B. i only 只有 i
C. ii only 只有 ii
D. i and ii i 和 ii

15. Consider the following program: 考慮以下程序:

Pascal	C	C++
<pre> var s: string; i, n, x, y: longint; begin read(s); n := length(s); y := 0; for i := 1 to n do begin x := ord(s[i]) - ord('A'); y := y * 16 + x; end; write(y); end. </pre>	<pre> char s[10]; int i, n, x, y; int main() { scanf("%s", s); n = strlen(s); y = 0; for (i = 0; i < n; i++) { x = (int)s[i] - (int)'A'; y = y * 16 + x; } printf("%d", y); return 0; } </pre>	<pre> string s; int i, n, x, y; int main() { cin >> s; n = s.length(); y = 0; for (i = 0; i < n; i++) { x = (int)s[i] - (int)'A'; y = y * 16 + x; } cout << y; return 0; } </pre>

For which of the following inputs will the program output an even number?

以下哪項輸入會令程序輸出一個雙數?

- i. BAD
- ii. CAFE
- iii. DEAF

- A. i only 只有 i
- B. ii only 只有 ii
- C. iii only 只有 iii
- D. i and iii only 只有 i 和 iii

16. Alice, Bob and Charlie are invited to a party. However, they still haven't decided whether they will go to the party or not. If Alice goes then Bob will also go; if Alice and Bob both go, then Charlie will not go. Which of the following is/are possible situations?

有人邀請愛麗絲、鮑伯和查理參加一個派對，但他們還未決定去不去。如果愛麗絲去，鮑伯也會一起去；如果愛麗絲和鮑伯都去，查理就不會去。以下哪些情況是有可能發生的？

- i. Only Alice and Bob go
只有愛麗絲和鮑伯去
- ii. Only Alice and Charlie go
只有愛麗絲和查理去
- iii. Only Bob and Charlie go
只有鮑伯和查理去

- A. i only 只有 i
- B. ii only 只有 ii
- C. i and iii only 只有 i 和 iii
- D. ii and iii only 只有 ii 和 iii

17. Consider the following program: 考慮以下程序：

Pascal	C	C++
<pre> var a, b, c, d, x, y: longint; begin a := 0; b := 0; c := 0; d := 0; read(x, y); if (x >= y) then a := 1; if (-x <= -y) then b := 1; if (y < x) then c := 1; if (not (x - y <= y - x)) then d := 1; // *** end. </pre>	<pre> int a, b, c, d, x, y; int main() { a = 0; b = 0; c = 0; d = 0; scanf("%d %d", &x, &y); if (x >= y) a = 1; if (-x <= -y) b = 1; if (y < x) c = 1; if (!(x - y <= y - x)) d = 1; // *** return 0; } </pre>	<pre> int a, b, c, d, x, y; int main() { a = 0; b = 0; c = 0; d = 0; cin >> x >> y; if (x >= y) a = 1; if (-x <= -y) b = 1; if (y < x) c = 1; if (!(x - y <= y - x)) d = 1; // *** return 0; } </pre>

Assume that the value of x and y are between $[-1000, 1000]$. At the moment that the program is executing the line marked with `***`, which variable(s) must have the same value as a ?

設 x 和 y 的值在 $[-1000, 1000]$ 之間，在程序運行以 `***` 標記的行時，哪些變量與 a 必定有相同的值？

- A. b only 只有 b
- B. c only 只有 c
- C. b and d only 只有 b 和 d
- D. c and d only 只有 c 和 d

18. x and y are 32-bit unsigned integers. Which of the expressions is/are always true?

x 和 y 是 32 位元 無符號整數；以下哪些表達式必定為真？

	Pascal	C / C++
i.	<input type="text" value="(x or y) >= (x and y)"/>	<input type="text" value="(x y) >= (x & y)"/>
ii.	<input type="text" value="(x or y) >= x"/>	<input type="text" value="(x y) >= x"/>
iii.	<input type="text" value="(x and y) <= x"/>	<input type="text" value="(x & y) <= x"/>

- A. None of the above 以上皆否
- B. i only 只有 i
- C. i and ii only 只有 i 和 ii
- D. i, ii and iii i, ii 和 iii

19. Consider the following program: 考慮以下程序：

Pascal	C	C++
<pre> var a, b, c: longint; begin read(a, b); if (a <= 10) then begin if (b >= 30) then c := 15 else if (a >= 5) then c := 40 else c := 35 end; if (b >= 20) then begin if (a >= 20) then c := 30 else if (b <= 30) then c := 20 else c := 25 end; write(c) end.</pre>	<pre> int a, b, c; int main() { scanf("%d %d", &a, &b); if (a <= 10) { if (b >= 30) c = 15; else if (a >= 5) c = 40; else c = 35; } if (b >= 20) { if (a >= 20) c = 30; else if (b <= 30) c = 20; else c = 25; } printf("%d", c); return 0; }</pre>	<pre> int a, b, c; int main() { cin >> a >> b; if (a <= 10) { if (b >= 30) c = 15; else if (a >= 5) c = 40; else c = 35; } if (b >= 20) { if (a >= 20) c = 30; else if (b <= 30) c = 20; else c = 25; } cout << c; return 0; }</pre>

Which of the following inputs generates the greatest output?

以下哪一個輸入會生成最大的輸出？

- A. 5 20
- B. 10 15
- C. 20 30
- D. 25 35

20. Consider the following program: 考慮以下程序：

Pascal	C	C++
<pre> var a, b, i, j: longint; begin read(a, b); for i := 1 to a do for j := 7 downto b do write('*') end.</pre>	<pre> int a, b, i, j; int main() { scanf("%d %d", &a, &b); for (i = 1; i <= a; i++) for (j = 7; j >= b; j--) printf("*"); return 0; }</pre>	<pre> int a, b, i, j; int main() { cin >> a >> b; for (i = 1; i <= a; i++) for (j = 7; j >= b; j--) cout << "*"; return 0; }</pre>

On which of the following input does the program output the largest number of *s?

以下哪一項輸入會令以上程序輸出最多 * ？

- A. 3 4
- B. 5 6
- C. 10 10
- D. 11 7

21. What is the output of the following program? 以下程序的輸出是？

Pascal

```
var
  i, j, t: longint;
  a: array[0..7] of longint =
    (1, 7, 5, 6, 4, 3, 8, 2);
begin
  for i := 0 to 6 do
    for j := 0 to 6 - i do
      if (a[i] < a[i + 1]) then
        begin
          t := a[i];
          a[i] := a[i + 1];
          a[i + 1] := t;
        end;
    for i := 0 to 7 do
      write(a[i], ' ');
end.
```

C

```
int i, j, t;
int a[8] =
  {1, 7, 5, 6, 4, 3, 8, 2};
int main() {
  for (i = 0; i <= 6; i++)
    for (j = 0; j <= 6 - i; j++)
      if (a[i] < a[i + 1]) {
        t = a[i];
        a[i] = a[i + 1];
        a[i + 1] = t;
      }
  for (i = 0; i <= 7; i++)
    printf("%d ", a[i]);
  return 0;
}
```

C++

```
int i, j, t;
int a[8] =
  {1, 7, 5, 6, 4, 3, 8, 2};
int main() {
  for (i = 0; i <= 6; i++)
    for (j = 0; j <= 6 - i; j++)
      if (a[i] < a[i + 1]) {
        t = a[i];
        a[i] = a[i + 1];
        a[i + 1] = t;
      }
  for (i = 0; i <= 7; i++)
    cout << a[i] << " ";
  return 0;
}
```

- A. 8 7 6 5 4 3 2 1
- B. 1 2 3 4 5 6 7 8
- C. 1 7 5 6 4 3 2 8
- D. 7 5 6 4 3 8 2 1

22. What is the output of the following program? 以下程序的輸出是？

Pascal

```
var
  score: longint;
begin
  score := 70;
  if (score >= 90) then
    write('A')
  else if (score >= 80) then
    write('B');
  if (score >= 70) then
    write('C');
  if (score >= 60) then
    write('D')
  else
    write('F')
end.
```

C

```
int score;
int main() {
  score = 70;
  if (score >= 90)
    printf("A");
  else if (score >= 80)
    printf("B");
  if (score >= 70)
    printf("C");
  if (score >= 60)
    printf("D");
  else
    printf("F");
  return 0;
}
```

C++

```
int score;
int main() {
  score = 70;
  if (score >= 90)
    cout << "A";
  else if (score >= 80)
    cout << "B";
  if (score >= 70)
    cout << "C";
  if (score >= 60)
    cout << "D";
  else
    cout << "F";
  return 0;
}
```

- A. C
- B. D
- C. CD
- D. CDF

23. Suppose there are n non-negative integers, a_1, a_2, \dots and a_n such that $a_1! + a_2! + \dots + a_n! = 2017$. When n is the minimum possible value, what is the minimum possible value of $a_1 + a_2 + \dots + a_n$? ($x!$ is the factorial of x)
假設有 n 個非負整數分別為 a_1, a_2, \dots 和 a_n 使得 $a_1! + a_2! + \dots + a_n! = 2017$ 。當 n 的取值為其最小可能值時， $a_1 + a_2 + \dots + a_n$ 的最小可能值為多少? ($x!$ 是 x 的階乘)

- A. 48
- B. 49
- C. 50
- D. 51

24. The following program is written in TypeScript, a programming language developed by Microsoft. Anders Hejlsberg, who created Turbo Pascal and is the lead architect of C#, is a core developer of this language.
以下程序是以 TypeScript 編寫，它是一個由微軟開發的編程語言。Turbo Pascal 發明家、C# 主設計師 Anders Hejlsberg 是此語言的主要開發者。

TypeScript

```
interface Student {  
  name: string;  
  data: [number];  
}  
function sortByName(a: Student[]): Student[] {  
  var result = a.slice(0);  
  result.sort((x, y) => {  
    return x.name.localeCompare(y.name);  
  });  
  return result;  
}
```

- Which of the following statement about `data` in the 3rd line of the program is true?
以下哪項關於程序中第三行 `data` 的陳述是正確的?

- A. The numerical values will be rounded to the nearest integer when stored
儲存時數值會取至最接近整數
 - B. It can only store a square number such as 0, 1, 4, 9, ...
只能儲存一個正方形數如 0, 1, 4, 9, ...
 - C. It can store an array of 10 numbers
可儲存一大小為 10 的數字陣列
 - D. `number` can be ignored to store a string
`number` 可被忽略從而儲存一字串
25. The purpose of `a.slice(0)` in the 6th line of the program is: (Hint: the first element of array `a` is `a[0]`)
程序中第六行 `a.slice(0)` 的目的為: (提示: 陣列 `a` 的首個元素為 `a[0]`)
- A. Obtain the first element of array `a` 讀取陣列 `a` 的第一個元素
 - B. Remove the first element from array `a`, then return `a` 從陣列 `a` 刪去第一個元素，然後傳回 `a`
 - C. Make sure that array `a` is not empty 確保陣列 `a` 並非空白
 - D. Make a copy of array `a` 複製一份陣列 `a`

END OF SECTION A 甲部完

Section B 乙部 (20 marks 分)

The blanks are labeled from A to M. Please fill in the blanks on the answer sheet.

下列各空格分別命名為 A 至 M，請在答題紙上對應的地方填上答案。

Note 注意：

- (1) Select exactly one programming language on the Answer Sheet. Answers must be in that language.
您必須在答題紙上選擇剛好一種編程語言，並只使用該語言作答。
- (2) For C and C++, you must not use the ?: operator.
對於 C 及 C++，答案不可以包括 ?: 運算元。
- (3) You must not use any library function unless the appropriate library has been included. (See Page 1)
除非適當的函數庫已被引用 (見頁一)，否則答案不可以包括任何函數庫內的函數。
- (4) You can write only one character in each box on the answer sheet.
答題紙上每個小格只可填上一個字符。
- (5) Answers must not exceed the designated number of boxes.
答案長度不得多於該題提供的小格數目。
- (6) Write legibly. Unrecognizable answers will be regarded as incorrect.
字體須端正清楚，無法辨別之答案當錯誤論。

1. Suppose array `int a[10]` (Pascal: `a: array[0..9] of longint`) contains 10 distinct integers. The program segment below tries to implement insertion sort to sort the array in ascending order. However, the program contains an error that can be fixed by changing exactly one line. Find the line and correct it. You may assume variables `i`, `j`, and `t` are declared as `int` (Pascal: `longint`).

現在一陣列 `int a[10]` (Pascal: `a: array[0..9] of longint`)，儲存著 10 個不相同的整數。以下程序段嘗試實現插入排序法來把陣列由小至大排序，但是程序中有一錯誤，並只需更改一行便能修正，請找出並將其改正。假設變量 `i`, `j` 和 `t` 已被宣告為 `int` (Pascal: `longint`)。

Pascal

```
11 for i := 1 to 9 do
12 begin
13   j := i - 1;
14   t := a[i];
15   while (j >= 0) do
16   begin
17     if (t > a[j]) then
18       break;
19     a[j + 1] := a[j];
20     j := j - 1
21   end;
22   a[j] := t
23 end;
```

C / C++

```
51 for (i = 1; i <= 9; i++)
52 {
53   j = i - 1;
54   t = a[i];
55   while (j >= 0)
56   {
57     if (t > a[j])
58       break;
59     a[j + 1] = a[j];
60     j = j - 1;
61   }
62   a[j] = t;
63 }
```

Line number 行數: A1

Correction 改正: A2 (2 marks 分)

Marks will only be given when both answers are correct.

兩個答案均為正確才會得分。

2. Reverse Polish notation is a kind of postfix notation, i.e. the operator is put on the right of the numbers. For instance, “1 + 2” will be written as “1 2 +” in reverse Polish notation. The algorithm of reverse Polish notation is fairly straightforward. Whenever there are n consecutive numbers before an operator (where n is the number of operands needed for the operator; for example, $n = 2$ for the operators “+”, “-”, “ \times ” and “ \div ”), we will replace the operator and the n operands with the calculated result.

For example: “5 1 2 + 4 \times + 3 -” \Rightarrow “5 3 4 \times + 3 -” \Rightarrow “5 12 + 3 -” \Rightarrow “17 3 -” \Rightarrow “14”

The above example is equivalent to “5 + (1 + 2) \times 4 - 3” in infix notation (i.e. the common notation we use). Furthermore, parentheses are not needed for reverse Polish notation as long as each operator has a fixed number of operands.

逆波蘭表示法是一種後綴表示法，即運算符會被放在數字的後面，例如“1 + 2”在逆波蘭表示法中會被寫成“1 2 +”。逆波蘭表示法的運算方法是挺直接的，當運算符的前面有連續 n 個操作數時（ n 是運算符所需的操作數數量；例如對於“+”、“-”、“ \times ”和“ \div ”這些運算符， $n = 2$ ），該運算符將會被運算，而該運算符和前面連續 n 個操作數將會被運算結果所取代。

例如：“5 1 2 + 4 \times + 3 -” \Rightarrow “5 3 4 \times + 3 -” \Rightarrow “5 12 + 3 -” \Rightarrow “17 3 -” \Rightarrow “14”

上述例子是等價於中綴表示法（即我們常用的表示法）中的“5 + (1 + 2) \times 4 - 3”。此外，如果每個運算符所需的操作數數量是不變的話，使用逆波蘭表示法是不需要加上任何括號的。

Evaluate the reverse Polish expression “7 5 + 3 \times 8 - 4 \div ”.

計算逆波蘭表達式 “7 5 + 3 \times 8 - 4 \div ”。

Answer 答案: B (1.5 marks 分)

Rewrite the infix expression “7 + 5 \times 3 - 8 \div 4” in reverse Polish notation where the ordering of numbers is the same as the original expression. Spaces between operators and operands are optional. (Hint: The above two expressions are not equivalent)

以逆波蘭表示法重寫中綴表達式 “7 + 5 \times 3 - 8 \div 4” 而數字的次序跟原表達式相同。可忽略運算符和操作數之間的空格。（提示：以上的兩項表達式並不等價）

Answer 答案: C (1.5 marks 分)

3. There is a row of 18 seats, numbered 1 to 18. Three people are already seated with seat numbers 4, 13, and 18 (each marked by a cross in the following diagram).

有 18 個座位排成一行，座號是 1 至 18。三個人已分別坐在 4、13 和 18 號位（在下圖以交叉標示）。



Now, three more people come. Where should they sit so that the minimum distance between any pair of people is maximal? Write down their seat numbers (in any order), as well as the maximal distance.

The distance between seat A and seat B ($A < B$) is $B - A$.

現在，又有三個人來了。他們應該坐在哪裡，使得坐得最相近的兩個人之間的距離最大？寫下他們的座號（次序不限），以及最大距離。

座位 A 與座位 B ($A < B$) 之間的距離是 $B - A$ 。

Seat numbers 座號: D1 , D2 , D3 (1 mark for all correct; 全對得 1 分)

Maximal distance 最大距離: E (1 mark 分)

4. A string is a palindrome if and only if it reads the same backward or forward. For example, abcba and adda are palindromes while abcab and abc are not. A substring of a string is a continuous segment of characters appearing in the given string. For example, a, ab, bc and abc are substrings of abc while ac is not. The function `f(st, ed)` verifies if the substring `s[st..ed]` is a palindrome. It should return `true` if `s[st..ed]` is a palindrome and returns `false` otherwise. Please complete the following function so that it can verify if a substring is a palindrome correctly. Assume that `st` and `ed` are valid positions of the string and `st <= ed`. 我們稱一個字串為迴文當且僅當把這字串按相反順序重新排列後，得到的字串和原來的字串一樣。例如 abcba 和 adda 是迴文，abcab 和 abc 則不是迴文。我們稱一個字串的任何一段連續的字為它的子字串。例如 a、ab、bc 和 abc 是 abc 的子字串，ac 則不是。函數 `f(st, ed)` 是用來檢驗 `s` 的子字串 `s[st..ed]` 是否一個迴文，如 `s[st..ed]` 是迴文，函數應傳回 `true`，在其他情況下傳回 `false`。完成函數令其能正確檢驗一個子字串是否迴文。假設 `st` 和 `ed` 均為合法的字串位置且 `st <= ed`。

Pascal	C	C++
<pre> var s: ansistring; function f(st, ed: longint): boolean; var i: longint; begin for i := st to ed do if (s[i] <> s[___F___]) then begin f := false; exit end; f := true end;</pre>	<pre> char s[100]; bool f(int st, int ed) { int i; for (i = st; i <= ed; i++) if (s[i] != s[___F___]) return false; return true; }</pre>	<pre> string s; bool f(int st, int ed) { int i; for (i = st; i <= ed; i++) if (s[i] != s[___F___]) return false; return true; }</pre>

Answer 答案: _____ F _____ (1 mark 分)

Implement another approach to verify if `s[st..ed]` is a palindrome. Complete the following function. 實作另一種方式去檢驗 `s[st..ed]` 是否迴文，完成以下函數。

Pascal	C	C++
<pre> var s: ansistring; function f(st, ed: longint): boolean; begin if (ed - st = 0) then f := ___G___ else if (ed - st = 1) then f := ___H___ else if (ed - st >= 2) then f := ___I___; end;</pre>	<pre> char s[100]; bool f(int st, int ed) { if (ed - st == 0) return ___G___; else if (ed - st == 1) return ___H___; else if (ed - st >= 2) return ___I___; }</pre>	<pre> string s; bool f(int st, int ed) { if (ed - st == 0) return ___G___; else if (ed - st == 1) return ___H___; else if (ed - st >= 2) return ___I___; }</pre>

Answer 答案: _____ G _____ (1 mark 分)

Answer 答案: _____ H _____ (1 mark 分)

Answer 答案: _____ I _____ (2 marks 分)

5. Consider the following program: 考慮以下程序:

Pascal

```
var
  a, val, i, j: longint;
begin
  read(a);
  for i := 1 to a do
    for j := 1 to a do
      begin
        val :=        J/K       ;
        write(val);
        if (j <> a) then
          write(' ')
        else
          writeln
      end
    end
  end.
```

C

```
int a, val, i, j;
int main() {
  scanf("%d", &a);
  for (i = 1; i <= a; i++)
    for (j = 1; j <= a; j++) {
      val =        J/K       ;
      printf("%d", val);
      if (j != a)
        printf(" ");
      else
        printf("\n");
    }
}
```

C++

```
int a, val, i, j;
int main() {
  cin >> a;
  for (i = 1; i <= a; i++)
    for (j = 1; j <= a; j++) {
      val =        J/K       ;
      cout << val;
      if (j != a)
        cout << " ";
      else
        cout << endl;
    }
}
```

Complete the program so that when input is 4, the program gives the output to the right.

完成程序使得當程序的輸入為 4 時, 程序會產生右方之輸出。

Answer 答案: J (2 marks 分)

```
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

Complete the program so that when input is 5, the program gives the output to the right.

完成程序使得當程序的輸入為 5 時, 程序會產生右方之輸出。

Answer 答案: K (2 marks 分)

```
1 2 3 4 5
2 3 4 5 1
3 4 5 1 2
4 5 1 2 3
5 1 2 3 4
```

6. A three-digit positive integer is a wonderful integer if its sum of digits equals 8. For example, 125 and 332 are wonderful integers, while 801 and 232 are not. Complete the function so that `is_wonderful(x)` returns true if `x` is a wonderful integer and false if otherwise. You may assume that `x` is a three-digit positive integer.

如果一個三位正整數的三個位加起來等於 8, 我們稱之為奇妙數。例如 125 和 332 是奇妙數, 而 801 和 232 不是。請你完成以下函數, 使得 `is_wonderful(x)` 在 `x` 是奇妙數時傳回 true, 否則傳回 false。你可假設 `x` 是一個三位正整數。

Pascal

```
function is_wonderful(x: longint): boolean;
begin
  is_wonderful :=        L       ;
end;
```

C / C++

```
bool is_wonderful(int x) {
  return        L       ;
}
```

Answer 答案: L (2 marks 分)

7. What is the output of the following program? 以下程序的輸出是？

Pascal

```
var
  a: array[0..7] of longint =
    (5, 7, 4, 3, 2, 6, 1, 8);
  i, j, r: longint;
begin
  r := 0;
  for i := 0 to 7 do
    for j := 0 to i - 1 do
      if (a[j] > a[i]) then
        r := r + 1;
  write(r)
end.
```

C

```
int a[8] =
  {5, 7, 4, 3, 2, 6, 1, 8};
int i, j, r;
int main() {
  r = 0;
  for (i = 0; i <= 7; i++)
    for (j = 0; j <= i - 1; j++)
      if (a[j] > a[i])
        r = r + 1;
  printf("%d", r);
  return 0;
}
```

C++

```
int a[8] =
  {5, 7, 4, 3, 2, 6, 1, 8};
int i, j, r;
int main() {
  r = 0;
  for (i = 0; i <= 7; i++)
    for (j = 0; j <= i - 1; j++)
      if (a[j] > a[i])
        r = r + 1;
  cout << r;
  return 0;
}
```

Answer 答案: _____ M _____ (2 marks 分)

END OF PAPER 全卷完