

Format 類型	# Questions 題目數	Total Marks 佔分
Section A 甲部	Multiple Choice 多項選擇題	25
Section B 乙部	Fill-in-the-blanks 填充題	8 (A - N)
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) Other than questions that mention compilation, assume all programs are compiled properly in Ubuntu 20.04 using the compilers and commands below.

除了有提及編譯的題目之外，假設所有程序都在 Ubuntu 20.04 下使用以下編譯器及指令正確地編譯。

Pascal: Free Pascal (<code>fpc-compiler 3.0.4</code>)	<code>fpc program.pas</code>
C: GNU GCC (<code>gcc-4.9 4.9.4</code>)	<code>gcc -std=c99 program.c -o program</code>
C++: GNU G++ (<code>g++-4.9 4.9.4</code>)	<code>g++ -std=c++98 program.cpp -o program</code>

Section A 甲部 (25 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。

1. Alice is about to conduct word frequency analysis on the Oxford Dictionary, before that Alice makes two guesses:
愛麗絲正準備對牛津字典進行字詞頻率分析，在此之前愛麗絲作出兩個猜測：

- i. There are more words containing 'e' than words containing 'te' or 'es'.
含有 'e' 的字詞比含有 'te' 或 'es' 的字詞要多。
- ii. There are more words ending with 'ing' than words with 'n' as the second last letter.
以 'ing' 為結尾的字詞比倒數第二個字母為 'n' 的字詞要多。

Which of the following is correct?

以下哪一項是正確的？

- A. Both Statements i and ii must be true.
陳述 i 和陳述 ii 都必然為真。
- B. Both Statements i and ii must be false.
陳述 i 和陳述 ii 都必然為假。
- C. Statement i must be true and Statement ii must be false.
陳述 i 必然為真，陳述 ii 必然為假。
- D. The correctness of at least one statement cannot be determined before the word frequency analysis is conducted.
至少一個陳述的真確性在進行字詞頻率分析前無從判斷。

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

Pascal

```
var n, i, x: longint;
  s: string = 'hkoi.org';
begin
  n := length(s);
  x := 0;
  for i := 1 to n do
    if (s[i] <> s[n - i + 1]) then
    begin
      s[i] := s[n - i + 1];
      inc(x)
    end;
  write(x)
end.
```

C

```
int n, i, x;
char s[] = "hkoi.org";
int main() {
  n = strlen(s);
  x = 0;
  for (i = 0; i < n; i++)
    if (s[i] != s[n - i - 1]) {
      s[i] = s[n - i - 1];
      x++;
    }
  printf("%d", x);
  return 0;
}
```

C++

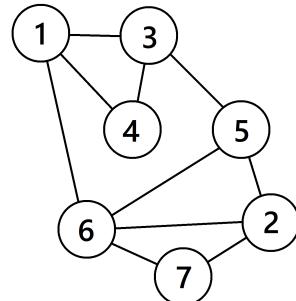
```
int n, i, x;
string s = "hkoi.org";
int main() {
  n = s.length();
  x = 0;
  for (i = 0; i < n; i++)
    if (s[i] != s[n - i - 1]) {
      s[i] = s[n - i - 1];
      x++;
    }
  cout << x;
  return 0;
}
```

- A. 3
- B. 4
- C. 6
- D. 8

3. Consider the undirected graph on the right. What is the minimum number of edges needed to be deleted so that node 1 and 2 are disconnected?

考慮右方所示的無向圖。最少需要刪除多少條邊才能使節點 1 和 2 互不相連？

- A. 2
- B. 3
- C. 4
- D. 5



4. If it is sunny on Sunday, Charlie goes cycling. If Charlie does not go cycling, he studies at home. If Charlie does not study at home, he fails the test. Given that Charlie has not failed the test, which of the following statements **may** be true?

如果星期日是晴天，查理就會去踩單車。如果查理沒有去踩單車，他就會在家中溫習。如果查理沒有在家中溫習，他測驗就會不合格。已知查理測驗沒有不合格，以下哪項陳述**可能**為真？

- i. Charlie has studied at home.
查理有在家中溫習。
- ii. Charlie has gone cycling.
查理有去踩單車。
- iii. It was not sunny on Sunday.
星期天不是晴天。

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

5. Which of the following Boolean expressions is not logically equivalent to other Boolean expressions?

以下哪個布爾表達式與其他布爾表達式在邏輯上不等價？

- A. A OR (NOT B OR NOT C)
- B. NOT (((NOT A) AND B) AND C)
- C. (A OR NOT B) OR NOT C
- D. ((NOT A) AND B) OR NOT C

6. Which of the following numbers are equal?

以下哪些數字是相等的？

- i. 93.85_{10}
 - ii. 1011101.11011001_2
 - iii. 135.6662_8
 - iv. $5C.C9_{16}$
- A. i and ii only 只有 i 和 ii
 - B. ii and iii only 只有 ii 和 iii
 - C. i, ii and iii only 只有 i、ii 和 iii
 - D. ii, iii and iv only 只有 ii、iii 和 iv

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

Pascal

```
var
  a: array[0..9] of longint =
  (3, 1, 4, 1, 5, 9, 2, 6, 5, 3);
  i, j, k, s, x: longint;
begin
  x := 0;
  for i := 0 to 9 do
    for j := i to 9 do
      begin
        s := 0;
        for k := i to j do
          s := s + a[k];
        if (s mod 2 = 0) then
          inc(x);
      end;
  write(x)
end.
```

C

```
int a[10] = {3, 1, 4, 1, 5, 9,
             2, 6, 5, 3};
int i, j, k, s, x;
int main() {
  x = 0;
  for (i = 0; i <= 9; i++)
    for (j = i; j <= 9; j++) {
      s = 0;
      for (k = i; k <= j; k++)
        s = s + a[k];
      if (s % 2 == 0)
        x++;
    }
  printf("%d", x);
  return 0;
}
```

C++

```
int a[10] = {3, 1, 4, 1, 5, 9,
             2, 6, 5, 3};
int i, j, k, s, x;
int main() {
  x = 0;
  for (i = 0; i <= 9; i++)
    for (j = i; j <= 9; j++) {
      s = 0;
      for (k = i; k <= j; k++)
        s = s + a[k];
      if (s % 2 == 0)
        x++;
    }
  cout << x;
  return 0;
}
```

- A. 25
- B. 26
- C. 27
- D. 28

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

Pascal

```
var
  a: array[0..6] of longint =
  (3, 6, 8, 2, 5, 1, 2);
function f(x: longint): longint;
begin
  if (x >= 4) then
    f := a[x]
  else
    begin
      a[x] := a[x] + f(x + 1);
      a[x] := a[x] + f(x + 2);
      f := a[x]
    end
  end;
begin
  write(f(0))
end.
```

C

```
int a[7] = {3, 6, 8, 2, 5, 1,
            2};
int f(int x) {
  if (x >= 4)
    return a[x];
  else {
    a[x] += f(x + 1);
    a[x] += f(x + 2);
    return a[x];
  }
}
int main() {
  printf("%d", f(0));
  return 0;
}
```

C++

```
int a[7] = {3, 6, 8, 2, 5, 1,
            2};
int f(int x) {
  if (x >= 4)
    return a[x];
  else {
    a[x] += f(x + 1);
    a[x] += f(x + 2);
    return a[x];
  }
}
int main() {
  cout << f(0);
  return 0;
}
```

- A. 59
- B. 84
- C. 87
- D. 90

9. Which of the following statements about Merge Sort and Quicksort is correct?

以下哪些對於合併排序和快速排序的描述是正確的？

- i. Merge sort does not need to compare the same two elements more than once.

合併排序無須比較相同的兩個元素多於一次。

- ii. Quicksort is a comparison-based sorting.

快速排序是比較排序的一種。

A. i only 只有 i

B. ii only 只有 ii

C. i and ii i 和 ii

D. None of them 無

10. Given an array $a[100]$ (Pascal: $a[0..99]$) of 100 distinct integers. After running the program segment below, which of the following statements must be true?

給定一個由 100 個不同的整數組成的數組 $a[100]$ (Pascal: $a[0..99]$)。在執行以下程序段之後，請問下列哪項陳述必定為真？

Pascal

```
for i := 0 to 99 do
  for j := 0 to 99 do
    if (a[i] < a[j]) then
      begin
        temp := a[i];
        a[i] := a[j];
        a[j] := temp
      end
```

C / C++

```
for (i = 0; i <= 99; i++)
  for (j = 0; j <= 99; j++)
    if (a[i] < a[j]) {
      temp = a[i];
      a[i] = a[j];
      a[j] = temp;
    }
```

- A. The array is sorted in ascending order

數組由小至大排列

- B. The array is sorted in descending order

數組由大至小排列

- C. $a[0]$ is the largest element of the array

$a[0]$ 是數組中最大的元素

- D. The array is reversed

數組被倒轉

11. Consider the directed graph on the right. If vertices and edges could be passed through repeatedly, which of the following cannot be the number of edges passed through walking from vertex 1 to 2?

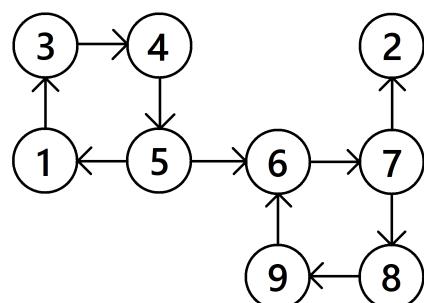
考慮右方所示的有向圖，若可以重覆經過節點及邊，由節點 1 走到節點 2，經過的邊的數目不可能是以下哪一項？

- A. 42

- B. 56

- C. 70

- D. 94



12. Alice and Bob are playing a game which requires them to flip some coins. They take turns to play until one wins. Alice moves first. In her turn she will flip a fair coin. She wins if she gets a head. Otherwise it becomes Bob's turn. In his turn he will flip a fair coin twice. He wins if both are heads. Otherwise it becomes Alice's turn again. What is the probability that Alice wins?

愛麗絲和鮑伯正在玩一個擲硬幣的遊戲。他們輪流玩這個遊戲，直至其中一人勝出為止。愛麗絲先開始。在她的回合中，她會擲一個公正硬幣。如果她擲到「公」，她會立即勝出，否則會輪到鮑伯的回合。在鮑伯的回合中，他會連續兩次擲一個公正硬幣。如果他擲到兩次「公」，他會立即勝出，否則會再次輪到愛麗絲的回合。

請問愛麗絲勝出這個遊戲的概率是多少呢？

- A. 50%
- B. 75%
- C. 80%
- D. 87.5%

13. Consider the following program segment with an unknown function $f(x)$ and a known function $g(x)$.

考慮以下程序段，當中包括一個未知函數 $f(x)$ 和一個已知函數 $g(x)$ 。

Pascal

```
function f(x: longint): boolean;
begin
  ... omitted 省略 ...
end;

function g(x: longint): boolean;
begin
  g := f(x) or not f(x)
end;
```

C / C++

```
bool f(int x) {
  ... omitted 省略 ...
}

bool g(int x) {
  return f(x) || !f(x);
}
```

Which of the following is/are true? 以下哪項為真？

- i. If f is not recursive, calling $g(x)$ once always calls $f(x)$ exactly twice.
如果 f 並不是遞歸，呼叫 $g(x)$ 一次總是呼叫 $f(x)$ 恰好兩次。
- ii. No matter what the value of x is, $g(x)$ always returns $true$.
不論 x 的值為何， $g(x)$ 總是傳回 $true$ 。

- A. i only 只有 i
- B. ii only 只有 ii
- C. i and ii i 和 ii
- D. None of them 無

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

Pascal

```

var
  stack: array[0..1, 0..9] of
    longint;
  t: array[0..1] of longint;
  i: longint;

procedure push(i, x: longint);
begin
  stack[i][t[i]] := x;
  t[i] := t[i] + 1
end;

procedure pop(i: longint);
begin
  t[i] := t[i] - 1
end;

function query(i: longint):
  longint;
begin
  query := stack[i][t[i] - 1]
end;

begin
  t[0] := 0;
  t[1] := 0;
  push(0, 10);
  push(0, 9);
  push(0, 1);
  push(0, 2);
  push(0, 3);
  push(1, 5);
  for i := 1 to 2022 do
  begin
    if (query(0) < query(1)) then
      begin
        push(1, query(0));
        pop(0)
      end
    else
      begin
        push(0, query(1));
        pop(1)
      end
  end;
  write(query(0), ' ', query(1))
end.

```

C

```

int stack[2][10];
int t[2];
int i;

void push(int i, int x) {
  stack[i][t[i]] = x;
  t[i] = t[i] + 1;
}

void pop(int i) {
  t[i] = t[i] - 1;
}

int query(int i) {
  return stack[i][t[i] - 1];
}

int main() {
  t[0] = 0;
  t[1] = 0;
  push(0, 10);
  push(0, 9);
  push(0, 1);
  push(0, 2);
  push(0, 3);
  push(1, 5);
  for (i = 1; i <= 2022; i++) {
    if (query(0) < query(1)) {
      push(1, query(0));
      pop(0);
    }
    else {
      push(0, query(1));
      pop(1);
    }
  }
  printf("%d %d", query(0),
         query(1));
  return 0;
}

```

C++

```

int stack[2][10];
int t[2];
int i;

void push(int i, int x) {
  stack[i][t[i]] = x;
  t[i] = t[i] + 1;
}

void pop(int i) {
  t[i] = t[i] - 1;
}

int query(int i) {
  return stack[i][t[i] - 1];
}

int main() {
  t[0] = 0;
  t[1] = 0;
  push(0, 10);
  push(0, 9);
  push(0, 1);
  push(0, 2);
  push(0, 3);
  push(1, 5);
  for (i = 1; i <= 2022; i++) {
    if (query(0) < query(1)) {
      push(1, query(0));
      pop(0);
    }
    else {
      push(0, query(1));
      pop(1);
    }
  }
  cout << query(0) << ' ' <<
       query(1);
  return 0;
}

```

- A. 1 2
- B. 2 1
- C. 3 5
- D. 9 1

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

Pascal

```
var a: array[0..14] of longint;
function aux(h, t, x: longint): longint;
var mid: longint;
begin
  if (h = t) then
  begin
    if (a[h] = x) then
      aux := h
    else
      aux := -1
  end
  else
  begin
    mid := (h + t) div 2;
    if (a[mid] < x) then
      aux := aux(mid + 1, t, x)
    else if (a[mid] > x) then
      aux := aux(h, mid - 1, x)
    else
      aux := mid
  end
end;
end;

function search(x: longint): longint;
begin
  search := aux(0, 14, x)
end;
```

C / C++

```
int a[15];

int aux(int h, int t, int x) {
  int mid;
  if (h == t) {
    if (a[h] == x)
      return h;
    else
      return -1;
  } else {
    mid = (h + t) / 2;
    if (a[mid] < x)
      return aux(mid + 1, t, x);
    else if (a[mid] > x)
      return aux(h, mid - 1, x);
    else
      return mid;
  }
}

int search(int x) {
  return aux(0, 14, x);
}
```

Assume $a[0] \dots a[14]$ contains 15 distinct integers between 1 and 100 inclusive, sorted in ascending order. How many times will function aux be called when executing the following program segment?

假設 $a[0] \dots a[14]$ 有 15 個 1 和 100 之間（含）不同的整數，而且從小至大排列。執行以下程序段時，函數 aux 會被呼叫多少次？

Pascal

```
for i := 1 to 100 do
  search(i);
```

C / C++

```
for (i = 1; i <= 100; i++)
  search(i);
```

- A. 374
- B. 389
- C. 474
- D. No sufficient information to determine. 資料不足而無法判斷。

16. Let a be a permutation of $\{1, 2, \dots, 10\}$. There are 10 chairs, numbered from 1 to 10, in a room. On each chair i , there is a sign with the number a_i . Now 10 students, also numbered from 1 to 10, enter the room. Initially, student i sits on chair i for all integers $1 \leq i \leq 10$. In each step, the person currently on chair i moves to chair a_i . All students move at the same time.

After N steps, for the first time, student i is on chair i for all integers $1 \leq i \leq 10$. If the students are allowed to choose the permutation a , what is the largest possible value of N ?

設 a 是 $\{1, 2, \dots, 10\}$ 的排列。房間裡有 10 張椅子，編號從 1 到 10。在每張椅子 i 上，都有一個帶有數字 a_i 的標誌。現在有 10 名學生進入房間，他們的編號也從 1 到 10。起初，對於所有整數 $1 \leq i \leq 10$ ，學生 i 都坐在椅子 i 上。在每一步，當前坐在椅子 i 上的學生會移動到椅子 a_i 。所有學生同時移動。在 N 步之後第一次發生對於所有整數 $1 \leq i \leq 10$ ，學生 i 都坐在椅子 i 上的情況。如果允許學生選擇排列 a ， N 的最大值是多少？

- A. 24
- B. 30
- C. 32
- D. 36

17. Consider a graph with 4 vertices. For each pair of vertices, there exists an undirected edge between them with probability $\frac{1}{2}$ independently. What is the probability that the graph contains only one cycle?

考慮一個含有 4 個頂點的圖，每對頂點之間獨立地有 $\frac{1}{2}$ 的概率存在一條無向邊。這個圖只包含一個環的概率是多少？

- A. $\frac{5}{64}$
- B. $\frac{7}{64}$
- C. $\frac{17}{64}$
- D. $\frac{19}{64}$

18. Consider indefinitely rolling a fair 12-face die, which its faces are numbered as 1 to 12. Which of the followings is/are true?

考慮不斷擲一個公平的 12 面骰子，且它的面被標籤為 1 至 12。以下哪項為真？

- i. The expected required number of rolls for the following event to occur is 12: the face numbered 1 appears for the first time.

要發生以下事件，所需擲骰子次數的期望值是 12：標籤為 1 的面出現第一次。

- ii. The probability of the following event to occur is 0.5: the face numbered 1 appears at least once in the first 6 rolls.

以下事件所發生的機率是 0.5：在首 6 次擲骰子時，標籤為 1 的面出現至少一次。

- A. i only 只有 i
- B. ii only 只有 ii
- C. i and ii i 和 ii
- D. None of them 無

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

Pascal

```
var
  x, y, i, n: longint;
  a: array[0..999999] of longint;
begin
  read(n);
  a[0] := n;
  x := 0;
  y := 1;
  while (x < y) do
  begin
    for i := 1 to a[x] do
    begin
      a[y] := a[x] - 1;
      inc(y);
    end;
    inc(x);
  end;
  write(y)
end.
```

C

```
int x, y, i, n;
int a[1000000];
int main() {
  scanf("%d", &n);
  a[0] = n;
  x = 0;
  y = 1;
  while (x < y) {
    for (i = 1; i <= a[x]; i++) {
      a[y] = a[x] - 1;
      y++;
    }
    x++;
  }
  printf("%d", y);
  return 0;
}
```

C++

```
int x, y, i, n;
int a[1000000];
int main() {
  cin >> n;
  a[0] = n;
  x = 0;
  y = 1;
  while (x < y) {
    for (i = 1; i <= a[x]; i++) {
      a[y] = a[x] - 1;
      y++;
    }
    x++;
  }
  cout << y;
  return 0;
}
```

What is the output of the program if the input is 2?

若輸入是 2，那麼程序的輸出是什麼？

- A. 3
- B. 4
- C. 5
- D. 6

20. Refer to the previous question, what is the output of the program if the input is 7?

承上題，若輸入是 7，那麼程序的輸出是什麼？

- A. 5040
- B. 13700
- C. 66320
- D. 873612

21. What is the number of different ways to represent 10 as summation of only integers 1, 2, 4, or 8? Two ways with the same integers in different order (for example, 1+1+8 and 8+1+1) are considered two different ways.

有多少種不同方法只以整數 1、2、4 或 8 組成的和表達式來表達 10？兩種以不同次序使用相同整數的方法（例如 1+1+8 與 8+1+1）視為兩種不同方法。

- A. 98
- B. 169
- C. 174
- D. 192

22. Consider a tree where all vertices either have degree 1 or degree 4. Which of the following statements must be true?

考慮一棵樹，它的所有頂點的度為 1 或 4。以下哪一個陳述必為真？

- i. The number of vertices with degree 1 is always even.
度為 1 的頂點數總是偶數。
- ii. The total number of vertices cannot be a multiple of 3.
頂點總數不能是 3 的倍數。

- A. i only 只有 i
- B. ii only 只有 ii
- C. i and ii i 和 ii
- D. None of them 無

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

Pascal

```
var
  a: array[0..4, 0..4] of
    longint = (
      (14, 9, 21, 1, 9),
      (3, 2, 8, 19, 6),
      (7, 0, 16, 15, 30),
      (4, 27, 31, 10, 11),
      (3, 4, 5, 6, 7));
  x, y, c, d: longint;
begin
  x := 0;
  y := 0;
  c := 0;
  d := 0;
  while ((x < 5) and (y < 5)) do
    begin
      c := c + a[x][y];
      d := 1 shl x;
      d := a[x][y] and d;
      if (d > 0) then
        x := x + 1
      else
        y := y + 1
    end;
    write(c)
end.
```

C

```
int a[5][5] = {
  {14, 9, 21, 1, 9},
  {3, 2, 8, 19, 6},
  {7, 0, 16, 15, 30},
  {4, 27, 31, 10, 11},
  {3, 4, 5, 6, 7}};
int x, y, c, d;
int main() {
  x = 0;
  y = 0;
  c = 0;
  d = 0;
  while ((x < 5) && (y < 5)) {
    c = c + a[x][y];
    d = 1 << x;
    d = a[x][y] & d;
    if (d > 0)
      x = x + 1;
    else
      y = y + 1;
  }
  printf("%d", c);
  return 0;
}
```

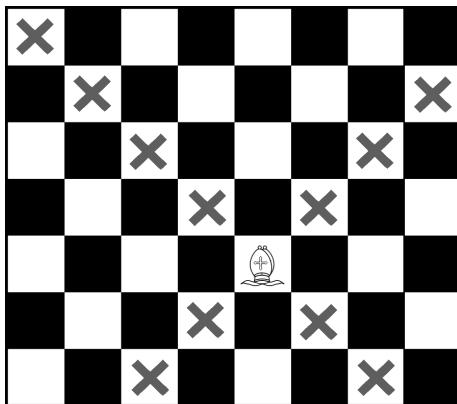
C++

```
int a[5][5] = {
  {14, 9, 21, 1, 9},
  {3, 2, 8, 19, 6},
  {7, 0, 16, 15, 30},
  {4, 27, 31, 10, 11},
  {3, 4, 5, 6, 7}};
int x, y, c, d;
int main() {
  x = 0;
  y = 0;
  c = 0;
  d = 0;
  while ((x < 5) && (y < 5)) {
    c = c + a[x][y];
    d = 1 << x;
    d = a[x][y] & d;
    if (d > 0)
      x = x + 1;
    else
      y = y + 1;
  }
  cout << c;
  return 0;
}
```

- A. 7
- B. 54
- C. 79
- D. 95

24. In Chess, bishops attack diagonally (see diagram below). At most how many bishops can be placed on a 7×8 chessboard, so that no two bishops attack one another?

國際象棋中，象（又稱主教）以斜行攻擊（見下圖）。請問在 7×8 的棋盤上，最多可擺放多少枚象，使得沒有兩枚象互相攻擊？



- A. 8
B. 12
C. 14
D. 15
25. Consider the following program segment: 考慮以下程序段：

Pascal

```
function f(x: longint): longint;
begin
  if (x = 1) then
    f := 1
  else
    f := f(x div 2) + f(x div 2)
      + ((x + 1) div 2) * ((x + 1) div 2)
end;
```

C / C++

```
int f(int x) {
  if (x == 1)
    return 1;
  else
    return f(x / 2) + f(x / 2)
      + ((x + 1) / 2) * ((x + 1) / 2);
}
```

What is the value of $f(32)$? 求 $f(32)$ 的值。

- A. 496
B. 512
C. 528
D. 639

END OF SECTION A 甲部完

Section B 乙部 (20 marks 分)

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

下列各空格分別命名為 A 至 N，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。
- (7) If blank X is divided into N parts X1, X2, ..., XN, it means that marks will only be given when X1, X2, ..., XN are all correct.
如果空格 X 分為 N 部份 X1、X2、...、XN，那麼 X1、X2、...、XN 皆為正確才會給分。

1. Complete the blanks below, so that $g(a, b)$ returns the value of a^b . It is guaranteed that $a, b > 0, a^b \leq 2^{31} - 1$.
完成以下程序，使得 $g(a, b)$ 的傳回值為 a^b 。題目保證 $a, b > 0, a^b \leq 2^{31} - 1$ 。

Pascal

```
function f(a, b, c: longint): longint;
begin
  if (b = 0) then
    f := _____ A1 _____
  else if ((b and 1) <> 0) then
    f := f(a * a, b div 2, a * c)
  else
    f := f(a * a, b div 2, _____ A2 _____)
end;

function g(a, b: longint): longint;
begin
  g := f(a, b, _____ A3 _____)
end;
```

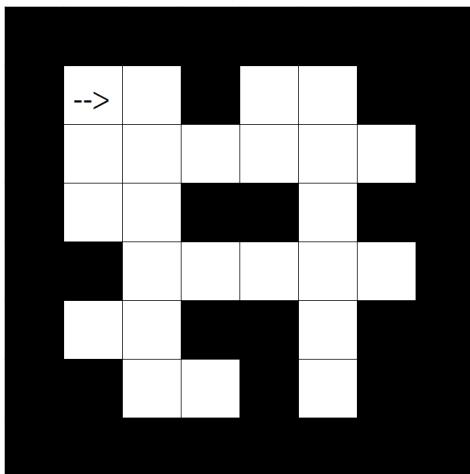
C / C++

```
int f(int a, int b, int c) {
  if (b == 0)
    return _____ A1 _____;
  else if ((b & 1) != 0)
    return f(a * a, b / 2, c * a);
  else
    return f(a * a, b / 2, _____ A2 _____);
}

int g(int a, int b) {
  return f(a, b, _____ A3 _____);
}
```

Answer 答案: _____ A1 _____ A2 _____ A3 _____ (2 marks 分)

2. Consider the following 8×8 maze. The arrow represents Alice. 考慮以下 8×8 迷宮。箭咀代表愛麗絲。



Alice is facing the direction that the arrow is pointing, and would like to walk in the maze. In each move, she may only perform either one of the following actions:

愛麗絲現面向箭咀指向的方向，並想在迷宮裏行走。每一步，她只可以進行以下其中一種操作：

1. Walk to the cell in front of her, or
向她前方的格子移動，或者
2. Rotate 90 degrees clockwise **and immediately** walk to the cell in front of her.
順時針旋轉 90 度然後立即向她前方的格子移動。

She may perform an action only if it does not make her walk into a black cell (she cannot perform the rotation only and not move forward). She can move to the cells that she has visited and the path may end at any cell. The starting cell is considered visited.

只有在該操作不會使她移動到黑色的格子時，她才能進行該操作（她不能只進行旋轉而不向前移動）。她可以向她已經到訪過的格子移動，而路徑可以在任何格子完結。起始的格子當作已到訪過。

A cell is reachable by Alice **if and only if** there exists a valid path along which Alice visited the cell. What is the number of cells that are reachable by Alice?

一個格子能夠被愛麗絲抵達當且僅當存在一個有效的路徑，在該路徑裏愛麗絲到訪過該格子。可以被愛麗絲抵達的格子數量是多少？

Answer 答案: _____ B _____ (1 mark 分)

What is the maximum number of **different** cells that Alice can visit in a **single** valid path? The starting cell is included.

在**單一條**有效的路徑裏，愛麗絲最多可以到訪多少個**不同的**格子？包含起始的格子。

Answer 答案: _____ C _____ (1 mark 分)

3. In a game, there is a game board of n rows and m columns, a monster is located on each cell. A warrior is located at (x, y) , meaning the x^{th} row and y^{th} column. He has two attack modes: s-attack and e-attack.

With s-attack, he can kill all monsters on the diagonal from top left to bottom right passing through (x, y) .

With e-attack, he can kill all monsters on the diagonal from top right to bottom left passing through (x, y) .

一個遊戲中，遊戲板有 n 行和 m 列格子，每一格上都有一隻怪物。有一個戰士身處 (x, y) ，即是第 x 行、第 y 列的格子。他有兩種攻擊模式：s-攻擊和 e-攻擊。

使用 s-攻擊時，他可以殺死所有身處從左上至右下且經過 (x, y) 的斜線的怪物。

使用 e-攻擊時，他可以殺死所有身處從右上至左下且經過 (x, y) 的斜線的怪物。

Refer to the following diagram. The warrior is located at $(2, 3)$, denoted as X.

With s-attack, He can kill all the monsters on cells denoted as S, including $(2, 3)$.

With e-attack, He can kill all the monsters on cells denoted as E, including $(2, 3)$.

參考下圖，戰士身處 $(2, 3)$ ，即有 X 標記的格子。

使用 s-攻擊時，他可以殺死所有身處有 S 標記的格子的怪物，包含 $(2, 3)$ 。

使用 e-攻擊時，他可以殺死所有身處有 E 標記的格子的怪物，包含 $(2, 3)$ 。

	1	2	3	4	5	6	7
1		S		E			
2			X				
3		E		S			
4	E				S		
5						S	

Pascal

```
procedure attack_s(n, m, x, y: longint);
var
  i, j: longint;
begin
  for i := 1 to n do
    for j := 1 to m do
      if (_____ D _____) then
        writeln(i, ',', j)
end;
```

C

```
void attack_s(int n, int m, int
  x, int y) {
  int i, j;
  for (i = 1; i <= n; i++)
    for (j = 1; j <= m; j++)
      if (_____ D _____)
        printf("%d,%d\n", i, j);
}
```

C++

```
void attack_s(int n, int m, int
  x, int y) {
  int i, j;
  for (i = 1; i <= n; i++)
    for (j = 1; j <= m; j++)
      if (_____ D _____)
        cout << i << "," << j <<
          endl;
}
```

Complete the function `attack_s` such that it outputs the coordinates of all monsters that the warrior can kill with s-attack, including (x, y) .

完成函數 `attack_s` 使其輸出所有可以被戰士使用 s-攻擊殺死的怪物的座標，包含 (x, y) 。

Answer 答案: _____ D _____ (1 mark 分)

Pascal

```
procedure attack_e(n, m, x, y:  
    longint);  
var  
    i, j: longint;  
begin  
    for i := 1 to n do  
        for j := 1 to m do  
            if (_____ E _____) then  
                writeln(i, ',', j)  
end;
```

C

```
void attack_e(int n, int m, int  
    x, int y) {  
    int i, j;  
    for (i = 1; i <= n; i++)  
        for (j = 1; j <= m; j++)  
            if (_____ E _____)  
                printf("%d,%d\n", i, j);  
}
```

C++

```
void attack_e(int n, int m, int  
    x, int y) {  
    int i, j;  
    for (i = 1; i <= n; i++)  
        for (j = 1; j <= m; j++)  
            if (_____ E _____)  
                cout << i << "," << j <<  
                endl;  
}
```

Complete the function `attack_e` such that it outputs the coordinates of all monsters that the warrior can kill with e-attack, including (x, y) .

完成函數 `attack_e` 使其輸出所有可以被戰士使用 e-攻擊殺死的怪物的座標，包含 (x, y) 。

Answer 答案: _____ E _____ (1 mark 分)

4. Alice and Peter are playing a game of numbers! Rules of the game are as follows:
小詩和小明正在玩數字遊戲！遊戲規則如下：

In the beginning of the game, each of Alice and Peter has an array of n integers. Alice and Peter take turns to remove an element from their array, with Alice being the first to move. The game ends when there is **one** element remaining in each of Alice's and Peter's array.

在遊戲開始時，小詩和小明各有一個長度為 n 的數組。小詩和小明輪流從各自的數組中刪除一個元素，小詩先行。當小詩和小明的數組中都剩餘 **一個**元素時，遊戲結束。

Alice tries to **maximise** the absolute difference of the 2 remaining elements, while Peter tries to **minimise** it. Suppose Alice and Peter both play optimally, find the absolute difference of the remaining elements.

小詩嘗試 **最大化**剩餘 2 元素的絕對差，而小明嘗試把它**最小化**。假設小詩和小明都依照最佳策略，請找出遊戲結束時剩餘元素的絕對差。

Given Alice's array is $\{2, 6, 7, 11\}$ and Peter's array is $\{3, 9, 13, 14\}$, what is the resulting absolute difference?
已知小詩的數組為 $\{2, 6, 7, 11\}$ ，小明的數組為 $\{3, 9, 13, 14\}$ ，請找出遊戲結束時剩餘元素的絕對差。

Answer 答案: _____ F _____ (1 mark 分)

Given Alice's array is $\{3, 4, 12, 19, 25, 34, 36, 37, 55, 66\}$ and Peter's array is $\{6, 9, 16, 23, 29, 31, 43, 54, 60, 70\}$, what is the resulting absolute difference?

已知小詩的數組為 $\{3, 4, 12, 19, 25, 34, 36, 37, 55, 66\}$ ，小明的數組為 $\{6, 9, 16, 23, 29, 31, 43, 54, 60, 70\}$ ，請找出遊戲結束時剩餘元素的絕對差。

Answer 答案: _____ G _____ (2.5 marks 分)

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

Pascal

```
11 function multiply(x, y: longint): longint;
12 var
13   ans: longint;
14 begin
15   ans := 0;
16   while (x > 0) do
17   begin
18     if (x mod 2 = 1) then
19       ans := ans + y;
20     x := x div 2
21   end;
22   multiply := ans
23 end;
```

C / C++

```
41 int multiply(int x, int y) {
42   int ans = 0;
43   while (x > 0)
44   {
45     if (x % 2 == 1)
46       ans = ans + y;
47     x = x / 2;
48   }
49   return ans;
50 }
51
52
53
```

The `multiply` function above attempts to calculate the product of two integers x and y . However, there is a bug in the function.

以上之 `multiply` 函數嘗試計算 x 及 y 的乘積，但此函數含有一錯誤。

Write down the return value of `multiply(2021, 2021)`.

寫下 `multiply(2021, 2021)` 的傳回值。

Answer 答案: _____ H _____ (1 mark 分)

The bug can be fixed by changing exactly one line. Find the line and correct it **WITHOUT** using asterisk (*) so that the function correctly returns the product xy for all integers $1 \leq x, y \leq 10^4$.

此錯誤只需更改一行便能修正。在**不使用星號(*)**的情況下，請找出並將其改正使得函數正確傳回所有整數 $1 \leq x, y \leq 10^4$ 中的乘積 xy 。

Line number 行數: _____ I1

Correction 改正: _____ I2 _____ (2 marks 分)

6. Consider the following functions: 考慮以下函數：

Pascal

```
function f(): longint;
begin
  .. omitted 省略 ..
end;
function g(): longint;
begin
  g := _____ J _____
end;
```

C / C++

```
int f() {
  .. omitted 省略 ..
}

int g() {
  return _____ J _____;
```

Function `f()` returns a random integer between 0 and 99 inclusive uniformly and independently. Complete function `g()` such that it returns a random integer between 0 and 79 inclusive uniformly and independently.

函數 `f()` 均等及獨立地隨機傳回一個 0 至 99 (含) 的整數。完成函數 `g()` 使其均等及獨立地隨機傳回一個 0 至 79 (含) 的整數。

Answer 答案: _____ J _____ (1.5 marks 分)

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

Pascal

```
function f(x: longint): longint;
var y: longint;
begin
  y := 0;
  if (x < 10) then
    f := x
  else
    begin
      while (x > 0) do
        begin
          y := y + x mod 10;
          x := x div 10
        end;
        f := f(y)
      end
    end;
end;
```

C / C++

```
int f(int x) {
  int y = 0;
  if (x < 10)
    return x;
  else {
    while (x > 0) {
      y = y + x % 10;
      x = x / 10;
    }
    return f(y);
  }
}
```

For how many integers x where $1 \leq x \leq 2021$ does $f(x)$ return 1?

有多少 $1 \leq x \leq 2021$ 中的整數 x 使得 $f(x)$ 傳回 1?

Answer 答案: _____ K _____ (1.5 marks 分)

The following program uses the function f above. What is the output of the program?

以下程序中使用了以上的函數 f ，程序的輸出是什麼？

Pascal

```
var
  a: array[0..19, 0..19] of
    longint;
  i, j, sum: longint;
begin
  for i := 0 to 19 do
  begin
    a[i, 0] := 1;
    for j := 1 to 19 do
      a[i, j] := 0
  end;
  sum := 0;
  for i := 1 to 19 do
    for j := 1 to i do
      a[i, j] := a[i - 1, j]
      + a[i - 1, j - 1];
  for i := 1 to 19 do
    sum := sum + a[19, i];
  write(f(sum))
end.
```

C

```
int a[20][20];
int i, j, sum;
int main() {
  for (i = 0; i <= 19; i++) {
    a[i][0] = 1;
    for (j = 1; j <= 19; j++)
      a[i][j] = 0;
  }
  sum = 0;
  for (i = 1; i <= 19; i++)
    for (j = 1; j <= i; j++)
      a[i][j] = a[i - 1][j]
      + a[i - 1][j - 1];
  for (i = 1; i <= 19; i++)
    sum = sum + a[19][i];
  printf("%d", f(sum));
  return 0;
}
```

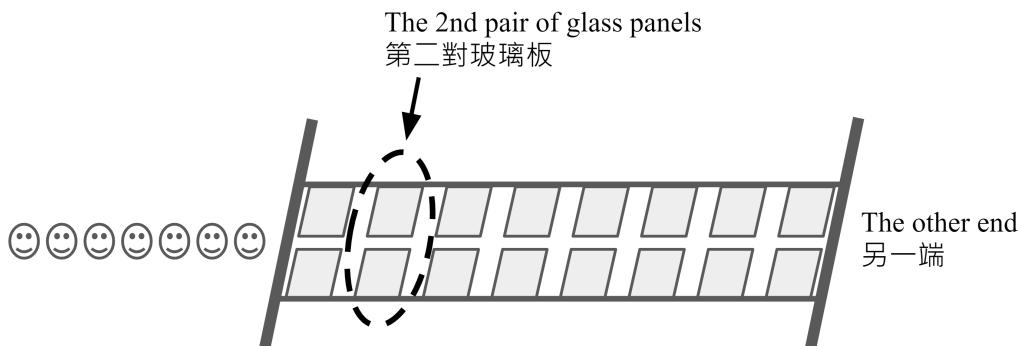
C++

```
int a[20][20];
int i, j, sum;
int main() {
  for (i = 0; i <= 19; i++) {
    a[i][0] = 1;
    for (j = 1; j <= 19; j++)
      a[i][j] = 0;
  }
  sum = 0;
  for (i = 1; i <= 19; i++)
    for (j = 1; j <= i; j++)
      a[i][j] = a[i - 1][j]
      + a[i - 1][j - 1];
  for (i = 1; i <= 19; i++)
    sum = sum + a[19][i];
  cout << f(sum);
  return 0;
}
```

Answer 答案: _____ L _____ (1.5 marks 分)

8. In a game, N participants, one by one in order, cross a bridge formed by K pairs of indistinguishable glass panels. At the beginning of the game, participants know nothing about how the glass panels are set up, except for the fact that: for every pair of glass panels, one of them is tempered glass which can support participant's weight, and another one is fragile glass which cannot.

在一個遊戲中， N 位玩家依序逐一橫過一條以 K 對無法分辨的玻璃板組成的橋。在遊戲開始時，玩家並不知道玻璃板是如何設定的，他們只知道：對於每對玻璃板，其中一塊為可以承托玩家重量的強化玻璃、另一塊為易碎玻璃則不可以。



An example of $N = 7, K = 8$ 的例子

In every step, a participant chooses one panel from the next pair to step on. When a participant steps on a fragile glass panel, the panel breaks and the participant gets eliminated. If steps on a tempered glass panel, the participant continues to the next step. A participant wins when arrives at the other end of the bridge. Assume all participants are smart and only aim for their personal win, and they can see which panels are broken and the decisions made by participants in front of them.

玩家每一步要從下一對玻璃中選擇踏上其中一塊。當一位玩家踏上易碎玻璃板，玻璃板碎裂而該玩家遭到淘汰。若踏上強化玻璃則繼續下一步。玩家成功到達橋的另一端則當作勝利。假設所有玩家皆是聰明且只以自己的勝利為目標，他們亦可以看到已碎裂的玻璃以及前方所有其他玩家的決定。

For the questions below, if the answer is not an integer, express the answer using reduced fraction.
對於以下問題，如答案非整數，請以最簡分數表示答案。

What is the expected number of winning participants when $N = 2$ and $K = 2$?

當 $N = 2$ 和 $K = 2$ 時，勝利玩家數目的期望值是甚麼？

Answer 答案: _____ M _____ (1 mark 分)

What is the expected number of winning participants when $N = 7$ and $K = 8$?

當 $N = 7$ 和 $K = 8$ 時，勝利玩家數目的期望值是甚麼？

Answer 答案: _____ N _____ (2 marks 分)

END OF PAPER 全卷完