HKOI 16/17 Junior

Section A Q7

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

Section A Q7

- What is the essence of recursion?
- What is the alternate method to implement Depth-First Search?
- **Stack**.

- You should not get wrong in this question in order to safely get into the final.

Section A Q14

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

Section A Q14

- For statement (i), notice the "**2 ***". It indicates the absolute fact that f(x) must be an even number, even *x* overflows or underflows.
  - What will *x* become if overflow or underflow occurs? How to represent the value of *x* undergoes overflow or underflow? Will they affect the parity of *x?*
- For statement (ii), it is trivially false due to overflow error.

- Study more related to the overflow issue of integers.

Section A Q24

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;
}
```

Section A Q24

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

Section A Q24

- What should we do towards a programming language that we don’t familiar with?
- Guess and observe.
- Why is it so special that the data type “number” is wrapped by brackets?
- Think of a scenario that we commonly use brackets: array!
- It is reasonable for us to guess that data is an integer array.

Section A Q25

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`

Section A Q25

- Observe more and guess.
- Consider the function name “sortByName”.
- Is it reasonable to only sort the first student’s name?
- Is it reasonable to sort only some of the students’ name?
- Is it reasonable to sort an empty list?
- Therefore, options A, B and C can be eliminated by observation.

- However, it is absolutely fine if you make wrong guesses.

Section B Q1

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`)。

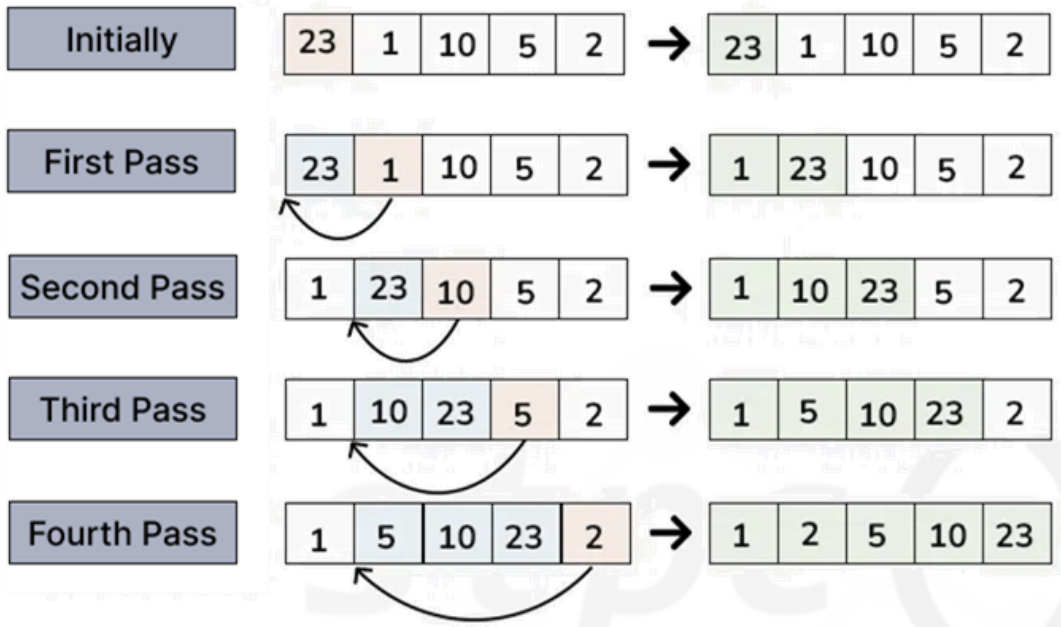
Section B Q1

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  }
```

Section B Q1

- What is insertion sort?
- Insert unprocessed elements to the sorted array one by one, until whole array has been sorted.
- We should take care of the swapping done while sorting
- After dry-running, we know that we should modify line 62 to `a[j+1] = t;`



Section B Q7

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

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;
}
```

Section B Q7

- The program actually count the number of pairs {i, j} such that a[j] > a[i] and j < i.
- By actually dry-running, the answer should be 16.

- Common mistake: 15 / 17
- Be careful!

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;
}
```