

2025 年度 算法入門 C++

第一次測驗

時間：2025 年 11 月 21 日 17:00 ~ 19:30

題目名稱	旅行計劃	高樓	回文字串	淘汰賽
題目類型	傳統型	傳統型	傳統型	傳統型
目錄	trip	buildings	string	knockout
輸入	標準輸入	標準輸入	標準輸入	標準輸入
輸出	標準輸出	標準輸出	標準輸出	標準輸出
每個測試點時限	1.0 秒	1.0 秒	2.0 秒	2.0 秒
每個測試點內存限制	256MB	256MB	512MB	512MB
子任務數量	2	5	5	3
樣例數量	4	6	3	4
提交處	trip	buildings	string	knockout

注意事項：(請仔細閱讀)

1. 本測驗使用 OI 賽制，學生提交後不會提供即時反饋，且對於每題，只會對該題的最後一次提交評分。
2. 時限為 150 分鐘。
3. 每題滿分為 100 分，總分 400 分。
4. 每題的分數由若干個子任務組成，學生的程序需通過子任務中的所有測試點才能獲得該子任務對應的分數。
5. C++ 中函數 `main()` 的返回值類型必須是 `int`，程序正常結束時的返回值必須是 0。
6. 若無特殊說明，結果的比較方式為全文比較（即過濾文末空格及文末行）。
7. 使用洛谷系統作答，請把程序上傳到對應的提交處。
8. 結束後，系統會顯示 0 分，這不是學生的成績。
9. 學生成績會在結束一週內公佈。

旅行計劃 (trip)

【題目描述】

小 G 將會帶着他的小貓到 A 國旅行。

小 G 計劃這次旅行會經過 n 座城市。小 G 對於第 i 座城市的好感度為 i ，而小貓對於第 i 座城市的好感度為 $a[i]$ 。

小 G 對該旅行的好感度是他對所經的 n 座城市的好感度總和。

同樣地，小貓對該旅行的好感度也是牠對所經的 n 座城市的好感度總和。

由於小 G 是極度愛貓之人，他希望小貓對旅行的好感度不比他的小，否則他就取消該旅行。於是找到了你，並要求你回答他的旅行計劃是否可行。

【輸入格式】

你需要在標準輸入 (`stdin`) 讀入數據。

輸入的第一行包含一個正整數 n 。

第二行包含 n 個整數 $a[1], a[2], a[3], \dots, a[n]$ ，以空格分隔。

【輸出格式】

你需要在標準輸出 (`stdout`) 輸出答案。

若計劃可行，輸出 YES；否則，輸出 NO。

【樣例 1 輸入】

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

【樣例 1 輸出】

```
NO
```

【樣例 1 解釋】

小 G 對於旅行的好感度為 $1 + 2 + 3 + 4 + 5 = 15$ 。

小貓對於旅行的好感度為 $2 + 4 + 3 + 1 + 2 = 12$ 。

小貓對旅行的好感度比小 G 的小，因此計劃不可行。

【樣例 2 輸入】

```
5
```

2 5 6 7 1

【樣例 2 輸出】

YES

【樣例 3】

見 `trip/trip3.in` 與 `trip/trip3.ans`。

這個樣例滿足子任務 1 的約束條件。

【樣例 4】

見 `trip/trip4.in` 與 `trip/trip4.ans`。

這個樣例滿足子任務 2 的約束條件。

【數據範圍】

對於所有測試數據，保證：

- $n \leq 1000$
- 對於所有滿足 $1 \leq i \leq n$ 的整數 i ， $-1000 \leq a[i] \leq 1000$ 。

子任務	分數	$n \leq$	$ a[i] \leq$	特殊性質
1	30	5	10	無
2	70	1000	1000	

高樓 (buildings)

【題目描述】

在美麗的澳門半島上，有 n 座高樓，第 i 座高樓的高度為 $h[i]$ 。

小 P 站在這 n 座高樓當中最高的一座上，俯瞰着整個澳門半島的美麗風景。

他看到了站在另一座高樓的小 Q，經過線上溝通後發現小 Q 在這 n 座高樓當中次高的一座上。其中，次高的高樓是指在所有高樓中，高度比最高的高樓小的高樓當中最高的一座。

小 P 想知道，他們所在的兩座高樓的高度之差是多少。

【輸入格式】

你需要在標準輸入 (`stdin`) 讀入數據。

輸入的第一行包含一個正整數 n 。

第二行包含 n 個正整數 $h[1], h[2], h[3], \dots, h[n]$ ，以空格分隔。

【輸出格式】

你需要在標準輸出 (`stdout`) 輸出答案。

輸出一行，一個整數表示答案。

【樣例 1 輸入】

```
5
7 4 8 8 3
```

【樣例 1 輸出】

```
1
```

【樣例 1 解釋】

小 P 站在最高的高樓（即第 3 座或第 4 座）上，其高度為 8。

小 Q 站在次高的高樓（即第 1 座）上，其高度為 7。

無論小 P 站在第 3 座還是第 4 座高樓上，兩高樓高度之差均為 $8 - 7 = 1$ 。

因此，答案為 1。

【樣例 2】

見 `buildings/buildings2.in` 與 `buildings/buildings2.ans`。

這個樣例滿足子任務 1 的約束條件。

【樣例 3】

見 buildings/buildings3.in 與 buildings/buildings3.ans。

這個樣例滿足子任務 2 的約束條件。

【樣例 4】

見 buildings/buildings4.in 與 buildings/buildings4.ans。

這個樣例滿足子任務 3 的約束條件。

【樣例 5】

見 buildings/buildings5.in 與 buildings/buildings5.ans。

這個樣例滿足子任務 4 的約束條件。

【樣例 6】

見 buildings/buildings6.in 與 buildings/buildings6.ans。

這個樣例滿足子任務 5 的約束條件。

【數據範圍】

對於所有測試數據，保證：

- $2 \leq n \leq 1000$
- 對於所有滿足 $1 \leq i \leq n$ 的整數 i ， $h[i] \leq 10^9$ 。
- 存在兩個滿足 $1 \leq i < j \leq n$ 的整數 i, j 使得 $h[i] \neq h[j]$ 。

子任務	分數	$n \leq$	特殊性質
1	5	10	B
2	25		無
3	15	1000	A
4	15		B
5	40		無

特殊性質 A：對於任意兩個整數 i, j ($1 \leq i < j \leq n$) 均有 $h[i] < h[j]$ 。

特殊性質 B：對於任意兩個整數 i, j ($1 \leq i < j \leq n$) 均有 $h[i] \neq h[j]$ 。

回文字串 (string)

【題目描述】

對於一個長度為 n 的字串 s ，如果該字串是一個回文字串，則對於所有整數 i ($1 \leq i \leq n$) 均有 $s[i] = s[n - i + 1]$ 。例如：`bob` 是一個回文字串，而 `agd` 不是。

小 A 有一個僅由英文字母組成的字串 s ，他希望把這個字串的所有大階英文字母都改為任意的小階英文字母，並令這個字串成為一個回文字串。

小 A 希望你幫他計算出，有多少個修改後的字串滿足這個要求。另外，小 A 怕這個數字太大，所以他讓你把這個答案對一個正整數 p 取模。其中，mod 是取餘運算， $a \bmod b$ (a 對 b 取模) 即為 a 除以 b 所得的餘數。

【輸入格式】

你需要在標準輸入 (`stdin`) 讀入數據。

本題有多組測試數據。

輸入的第一行包含一個整數 c ，表示子任務編號。 $c = 0$ 表示該測試點為樣例。

第二行包含一個正整數 T ，表示測試數據組數。

對於每組測試數據：

- 第一行包含一個字串 s 。
- 第二行包含一個正整數 p 。

【輸出格式】

你需要在標準輸出 (`stdout`) 輸出答案。

對於每組測試數據：

- 輸出一行，一個整數表示答案。

【樣例 1 輸入】

```
0
3
abACa
10
Abaka
10
aBBsbA
10
```

【樣例 1 輸出】

```
6  
0  
1
```

【樣例 1 解釋】

對於第一組測試數據，滿足要求的字串有：`ababa`、`abbba`、`abcba`、...、`abzba`，共 26 個字串滿足要求。對 $p = 10$ 取餘後應輸出 6。

對於第二組測試數據，沒有字串滿足要求，答案為 0。

對於第三組測試數據，滿足要求的字串只有 `abssba`，答案為 1。對 $p = 10$ 取餘後應輸出 1。

【樣例 2】

見 `string/string2.in` 與 `string/string2.ans`。

這個樣例滿足子任務 4 的約束條件。

【樣例 3】

見 `string/string3.in` 與 `string/string3.ans`。

這個樣例滿足子任務 5 的約束條件。

【數據範圍】

記 n 為字串 s 的長度。

對於所有測試數據，保證：

- $T \leq 10$
- $n \leq 10^6$
- $p \leq 10^6$
- 字串 s 僅由英文字母組成。

子任務	分數	$n \leq$	特殊性質
1	5	5	A
2	20		無
3	5	10^6	A
4	10		B
5	60		無

特殊性質 A：字串 s 僅由小階英文字母組成。

特殊性質 B：字串 s 僅由大階英文字母組成。

淘汰賽 (knockout)

【題目描述】

小 D 組織了 2^n 個人參加一場淘汰賽，他們的編號依次為 $1, 2, 3, \dots, 2^n$ 。編號為 i 的參賽者的武力值為 $a[i]$ 。

首先，他們會按編號順序排成一列，設當前有 $2k$ 位參賽者，則第 i (i 為奇數) 位參賽者會攻擊第 $i + 1$ 位參賽者。若第 i 位參賽者的武力值不小於第 $i + 1$ 位參賽者的武力值，第 $i + 1$ 位參賽者會被淘汰，否則第 i 位參賽者會被淘汰。這個過程稱作一輪淘汰。完成一輪淘汰後會剩下 k 位參賽者。

淘汰賽一共會進行 n 輪淘汰，最後會剩下一位參賽者，則這位參賽者勝出。

小 D 想知道，這 n 輪淘汰的參賽者的編號依次是什麼。

【輸入格式】

你需要在標準輸入 (`stdin`) 讀入數據。

輸入的第一行包含一個正整數 n 。

第二行包含 2^n 個正整數 $a[1], a[2], a[3], \dots, a[2^n]$ ，以空格分隔。

【輸出格式】

你需要在標準輸出 (`stdout`) 輸出答案。

輸出 n 行，第 i 行表示第 i 輪淘汰的參賽者的編號順序，以空格分隔。

【樣例 1 輸入】

```
2
2 4 1 5
```

【樣例 1 輸出】

```
1 3
2
```

【樣例 2】

見 `knockout/knockout2.in` 與 `knockout/knockout2.ans`。

這個樣例滿足子任務 1 的約束條件。

【樣例 3】

見 knockout/knockout3.in 與 knockout/knockout3.ans。

這個樣例滿足子任務 2 的約束條件。

【樣例 4】

見 knockout/knockout4.in 與 knockout/knockout4.ans。

這個樣例滿足子任務 3 的約束條件。

【數據範圍】

對於所有測試數據，保證：

- $n \leq 20$
- 對於所有滿足 $1 \leq i \leq 2^n$ 的整數 i ， $a[i] \leq 10^9$ 。

子任務	分數	$n \leq$	特殊性質
1	10	1	無
2	30	5	
3	60	20	