

第七屆高一程式設計排名賽 題本

注意事項

1. 本次比賽共 8 題，題本一共 21 頁。
 2. 比賽時間為 12:30 ~ 16:30，共 4 小時。
 3. 本次題目類型包含「標準輸入輸出」及「互動題」。
 4. 競賽中不得與他人交談，有問題請舉手詢問監試人員。
 5. 題目相關問題請透過 cms 競賽系統提問。
 6. 上廁所請舉手告知監試人員後方可離場。
 7. 不得做出干擾競賽之行為，違者主辦單位得取消其參賽資格。
 8. 最後一小時將凍結記分板，結果於賽後公告。
 9. 題目未依照難度排序，第 8 題為互動題。
-

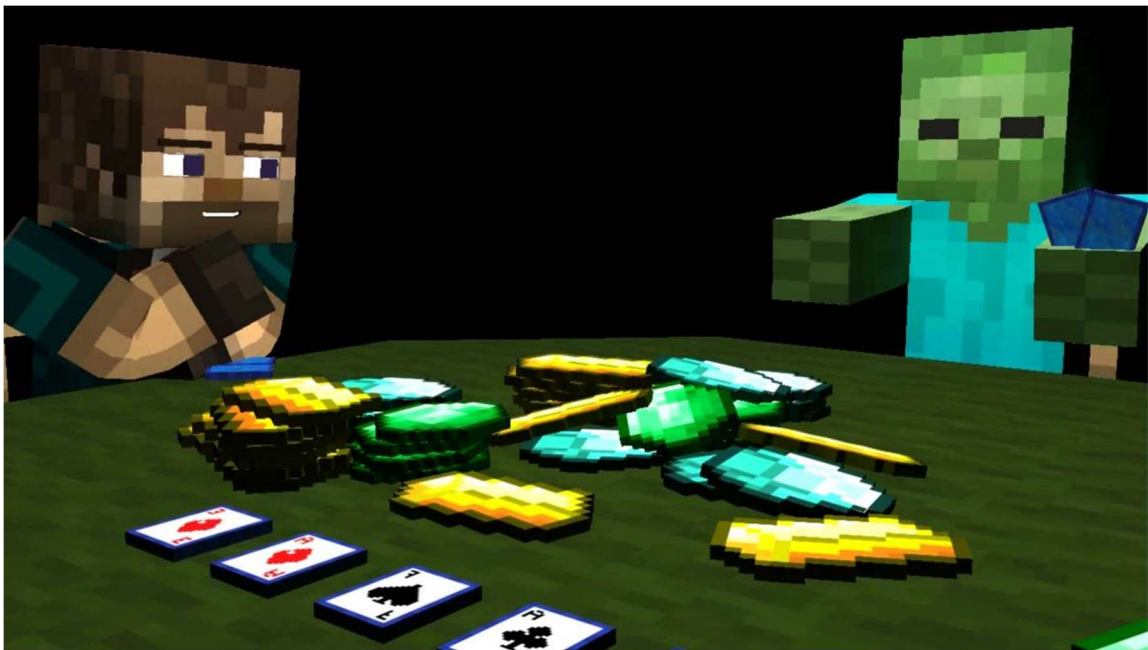
第一題：撲克 *Poker*

題目格式：標準輸入輸出

時間限制：0.1 秒

記憶體限制：5 MB

作者：台南一中 / 李長諺



題目敘述

Player 是一位浪遊四海、無所拘束(居無定所、沒有工作)的 Minecraft 冒險者。因為沒有住處，平常他都把所有東西放在身上，帶著旅行。

一天，Player 投宿在綠寶石小鎮的一間旅店。隔天醒來，他發現他被搶了！把整間房間翻地亂七八糟，也只找到 2 枚掉在牆角的銅幣。想要去找老闆抗議的他，看見老闆身旁兩隻巨大的鐵巨人保鏢正盯著自己，嚇得馬上離開。

Player 傷心地離開了酒店。走在街道上，正煩惱接下來要怎麼辦時，一個大大的招牌出現在他眼前。「發大財賭場！」。他決定了，打算進去賭一把，不成功，便成仁！他進到賭場，走向其中一張賭桌。

規則是，玩家跟莊家各抽一張撲克牌，只要出的牌比莊家大，就能獲勝；反之，輸掉這局。

撲克牌的大小：先比數值，同數值時比花色。黑桃 > 愛心 > 方塊 > 梅花。

他決定善用他靈巧的手(道具欄)來幫他贏得遊戲，他能夠迅速的換掉手上的牌，藉此獲勝。但是，為了避免因為一直出大牌而被人懷疑，他決定出一張剛好比莊家大一點的牌，於是請一旁的你幫忙。

請你在看到莊家的牌後，告訴他應該要出什麼牌。

輸入說明

輸入兩個以空白隔開的字元 A、B，代表莊家牌的花色、數字。

A 是 {S, H, D, C} 其中之一，分別對應 {黑桃, 愛心, 方塊, 梅花}。

B 是 {A, 2, 3, 4, 5, 6, 7, 8, 9, T, J, Q, K} 其中之一，分別對應 1~13。

輸出說明

如果該張牌不存在，請輸出 “IMPOSSIBLE”。

輸出兩個以空白隔開的字元 A、B，代表比莊家剛好大一點的牌。

格式請見 「輸入說明」。

子任務表

編號	分數	子任務內容
1	7 %	花色不包含黑桃
2	93 %	題目範圍

範例

範例輸入	範例輸出
C A	D A
D 7	H 7
H Q	S Q
S T	C J
S K	IMPOSSIBLE

第二題：重組回文 *Spell*

題目格式：標準輸入輸出

時間限制：1.0 秒

記憶體限制：512 MB

作者：台南一中 / 周暘典



題目敘述

在坑完發大財賭場的錢後，Player 賺到了足夠的錢，買下一棟別墅，以免住在外頭又被洗劫一空。

某天晚上，他準備出門打怪，突然發現他一直忘了做裝備。憧憬戰鬥的他，二話不說立刻開始動手。終於做好了武器跟鎧甲，準備進行附魔。

當他附魔時，他發現被附魔的道具會出現一段神秘文字(咒語)，而且附魔的咒語似乎會影響附魔的品質。經過無數次的研究和失敗後，他找出了強化附魔的方法和規律！

把附完魔的道具再放到附魔台上，花費一點經驗值，就能用羽毛筆寫上一個魔法文字。

而魔法文字有以下特性：

1. 已經寫上去的魔法文字(原咒語)不能去除。
2. 改變咒語時，只能在咒語尾端新增文字，否則附魔會失效。

3. 為了提升咒語的效力，新增的咒語應由原咒語從頭複製一段任意長度的字串重新排列而成。
4. 當咒語從前後讀起來都一樣，效果會達到最高級。

Player 想把身上的裝備都升到最高級，然而寫上的魔法文字越多，經驗值耗費越大。所以他想知道附完魔至少需要多少經驗值。

請你告訴他某段咒語要強化到最高級至少需要幾點經驗值。

輸入說明

第一行有一個數字 N ，代表咒語的長度。

第二行有一個字串 S ，代表咒語。

輸出說明

輸出一個數字，代表至少需要的經驗值。

子任務表

必定符合 $1 \leq N \leq 10,000$ ， S 只由大寫英文字母組成。

編號	分數	子任務內容
1	7 %	所有字元皆不相同
2	18 %	$N \leq 8$
3	18 %	存在某一個 j 符合 $a_1 \sim a_j$ 相等， $a_j \neq a_{j+1}$ ，且 $a_{j+1} \sim a_n$ 也相同
4	23 %	存在某一個 j 符合 $a_i \neq a_{j+1}$ ($1 \leq i \leq j$)，且 $a_{j+1} \sim a_n$ 相同
5	34 %	題目範圍

範例

範例輸入	範例輸出
12 SMAQNJXPXGDVY	11
2 AA	0
8 AAABAABA	2

第三題：賭博 *Gamble*

題目格式：標準輸入輸出

時間限制：1.0 秒

記憶體限制：512 MB

作者：台南一中 / 成俊宏



題目敘述

自從 Player 來到這座小鎮，也陸陸續續跟鎮民們有交流。

其中有一個人，名叫 Villager，是個賭鬼。自從他看到 Player 在賭場大展身手後，便三不五時跑去找他賭，久而久之也跟他變成了朋友。

有時候他們賭著賭著，發現金額太大了，於是想到用其他代幣來進行遊戲。他們決定把當天第一局的結果訂為 1 綠寶石。

遊戲結束後，兩人開始結算今天的結果。基於強迫症玩家的心態，Player 想知道今天的勝負能不能全部改由綠寶石交付，請你幫他算算看。

輸入說明

輸入兩個數字 A, B ，代表今天的勝負和 1 綠寶石代表的金額。

輸出說明

如果能全部改由綠寶石交付，輸出 “Yes”，否則輸出 “No”。

子任務表

必定符合 $-10^{100} \leq A \leq 10^{100}$ ， $-10^{16} \leq B \leq 10^{16}$ 。

編號	分數	子任務內容
1	5 %	$A = 0$ 或 $B = 1$
2	6 %	$-10^9 \leq A \leq 10^9$ ， $-10^8 \leq B \leq 10^8$
3	7 %	$-10^{18} \leq A \leq 10^{18}$
4	27 %	$A, B \geq 0$
5	23 %	$B = \{2, 3, 4, 5, 6, 8, 9\}$ 其中之一
6	32 %	題目範圍

範例

範例輸入	範例輸出
206248266 14880787	No
151288191792937066 1238171596257386	No
-115062172 -28765543	Yes

第四題：停止爆炸 *Explosion*

題目格式：標準輸入輸出

時間限制：1.0 秒

記憶體限制：512 MB

作者：台南一中 / 周暘典



題目敘述

Player 每天愜意地冒險、戰鬥、賭博(?)，過著愉快的生活。

然而對怪物們來說就不是那麼愉快了。身為戰鬥狂的 Player 的出現簡直是場災難。於是他們召開第 128 次怪物大會，討論該怎麼解決。

殭屍：「Errrr...」

苦力怕：「Sssss...」

骷髏：「(骨頭撞擊聲)」

...

與此同時，苦力怕家的兒子阿帕正在四處亂跑。走著走著突然看見熟悉的身影。是 Player！一

瞬間阿帕就被嚇得躲進了旁邊的草叢。

但過了很久，發現 Player 一動也不動，阿帕心想這是個好機會，就趁這次一口氣解決掉他。

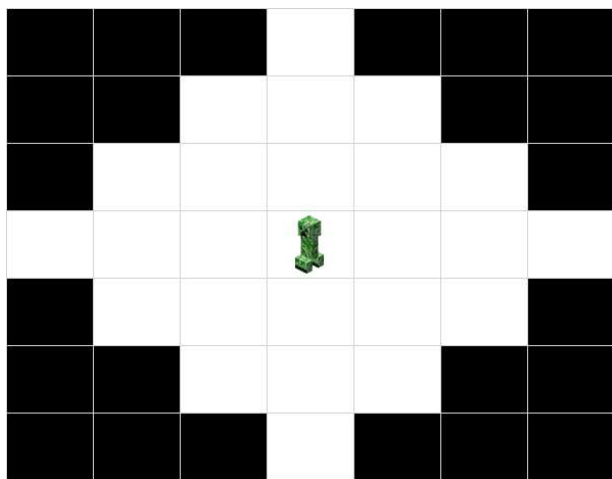
但是阿帕忘了考慮地形阻隔的問題，為了幫助阿帕的雄心壯志，給你現在這個區域的地形圖，請你替他算算他能不能成功炸到 Player。

阿帕的行動規則：

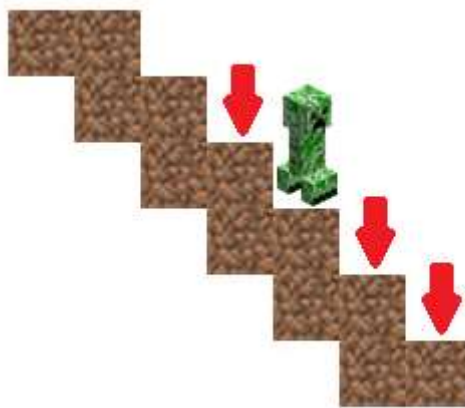
阿帕每次能上下左右移動一格的距離，但是他沒辦法跳太高，也不能向下掉太高的距離，不然他會摔倒起不來，所以他只能移動到高度比目前高 1 格或低 2 格以內的格子。

當阿帕要引爆時，他能炸到曼哈頓距離 3 以內的格子。

曼哈頓距離 = $| \text{目前的}x - \text{目標的}x | + | \text{目前的}y - \text{目標的}y |$



圖示 2: 引爆的範圍(曼哈頓距離 3 以內)



圖示 1: 行走時的高度差限制(箭頭代表可走)

輸入說明

第一行有兩個數字 N, M ，代表阿帕所在地的大小。

接下來 N 行之中，每行有 M 個數字，代表整張地圖。

接下來兩個數字 Cx, Cy ，代表阿帕的所在位置。

接下來兩個數字 Px, Py ，代表 Player 的所在位置。

輸出說明

如果阿帕炸的到 Player，輸出 “yes”，否則輸出 “no”。

子任務表

必定符合 $0 \leq A_{ij} \leq 255$ ， $1 \leq N, M \leq 1,000$ 。

編號	分數	子任務內容
1	3 %	$A_{ij} = 0$
2	13 %	$N = 1$
3	26 %	$A_{ij} = 0$ or 3
4	19 %	$1 \leq N, M \leq 5$
5	39 %	題目範圍

範例

範例輸入	範例輸出
5 5 0 0 0 0 0 3 3 3 3 0 3 3 3 3 0 3 3 3 3 3 3 3 3 3 3 1 1 5 5	yes
3 3 0 2 4 2 4 6 4 6 8 1 1 3 3	no
3 3 0 2 4 2 4 6 4 6 8 3 3 1 1	yes

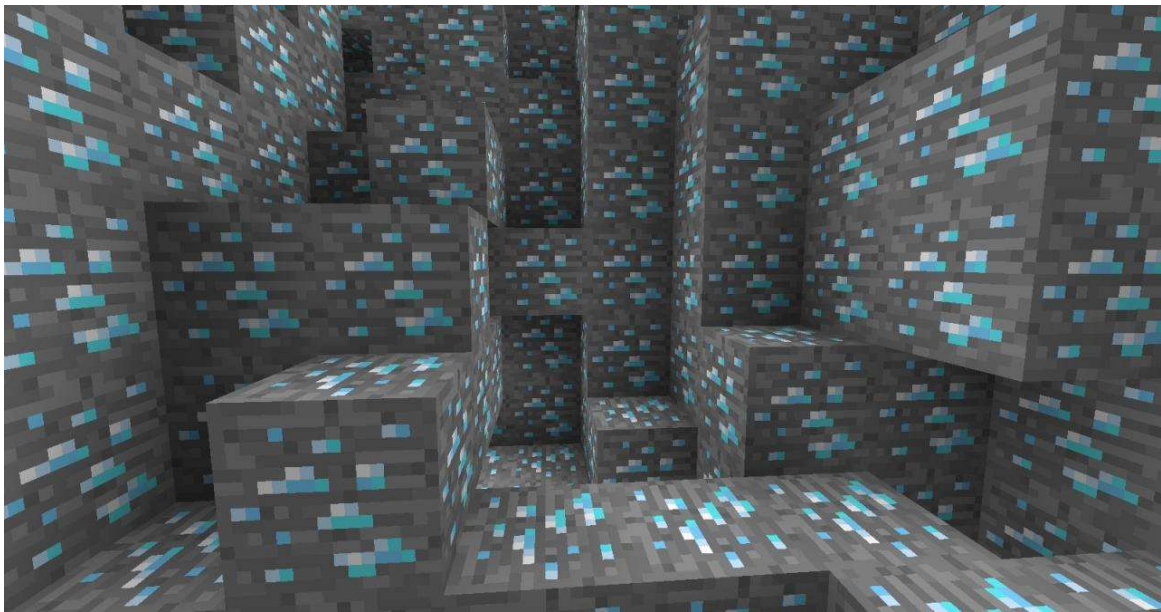
第五題：挖礦小隊 *Mining*

題目格式：標準輸入輸出

時間限制：1.0 秒

記憶體限制：512 MB

作者：台南一中 / 李長諺



題目敘述

在小鎮逐漸興振下，鎮民們發現了一個很嚴重的問題——鑽石快用完了。

為此，他們組成了挖鑽石小隊，準備下礦去挖鑽石。但每次要挖的時候，都會發生一些離奇的事件，比如鑽石下面突然冒出岩漿，或是在運送途中被絆倒，鑽石滾到峽谷裡等令人匪夷所思的事。

所以這群(沒用的)鎮民跑來找 Player 幫忙。在仔細調查完每個人後，他發現每個人都有所謂的幸運值。幸運值總和越高的小隊挖到的鑽石越多，但是如果一個小隊最幸運的人和最不幸的人幸運值差太大(超過某個數字 K)，運氣就會彼此衝突，發生一些衰事。

現在鎮民打算重新組挖礦小組，為了最大化挖礦的效益，請你幫鎮民們挑選適合的隊員。

輸入說明

第一行有兩個數字 N, K ，分別代表鎮民人數和數字 K 。
 接下來有 N 個數字 a_i ，代表每個鎮民的幸運值。

輸出說明

輸出一個數字，代表最大的幸運值總和。

子任務表

必定符合 $1 \leq N \leq 200,000$ ， $0 \leq k \leq 10^{12}$ ， $0 \leq a_i \leq 10^5$ 。

編號	分數	子任務內容
1	13 %	$k > (\text{最大的 } a_i) - (\text{最小的 } a_i)$ 或者 $k = 0$
2	15 %	$a_{i-1} \leq a_i$ ， $1 \leq N \leq 10,000$
3	31 %	$a_{i-1} \leq a_i$ ， $1 \leq N \leq 200,000$
4	41 %	題目範圍

範例

範例輸入	範例輸出
5 0 3 1 2 2 5	5
7 10 5 8 2 1 3 5 7	31
10 3 3 1 4 3 5 3 4 3 9 9	25

第六題：狗狗大隊 *Dogs*

題目格式：標準輸入輸出

時間限制：1.0 秒

記憶體限制：512 MB

作者：台南一中 / 周暘典



題目敘述

經過第 128 屆怪物大會後，怪物們決定派出骷髏弓箭手大隊去討伐 Player。

另一方面，知道怪物們行動的 Player，決定派出狗狗大隊去對付他們，於是他找到了 N 隻狗狗，組成獵犬隊。

然而組獵犬隊的時候有一些規則：

1. 每隻狗狗都有相對應的編號和戰鬥力
2. 狗狗可以被安排加入一個小隊，也可以不被安排進任何小隊
3. 可以選出多個小隊，每個小隊的狗狗編號必須是連續的
4. 小隊中的狗狗數量必須為 K ，不然他們會打架
5. 編號相鄰的狗狗不能分在不同小隊

現在，請你幫幫 Player，計算獵犬隊的戰鬥力最高總和是多少。

輸入說明

第一行有兩個數字 N, K ，分別代表狗狗大隊的人數和數字 K 。

接下來有 N 個數字，代表編號 $1 \sim N$ 的狗狗的戰鬥力。

輸出說明

輸出一個數字，代表戰鬥力最高總和。

子任務表

必定符合 $1 \leq K \leq N \leq 200,000$ ， $0 \leq a_i \leq 10^9$ 。

編號	分數	子任務內容
1	15 %	$2K > N$
2	16 %	$a_{i-1} \leq a_i$
3	20 %	$1 \leq K \leq N \leq 7$
4	21 %	$1 \leq N \leq 100,000$ ， $K = 1$
5	28 %	題目範圍

範例

範例輸入	範例輸出
5 4 2 3 3 4 3	13
7 1 1 4 1 2 5 2 5	14
7 3 1 2 3 3 2 1 2	11

第七題：246 三硝基甲苯 TNT

題目格式：標準輸入輸出

時間限制：5.0 秒

記憶體限制：512 MB

作者：竹科實中 / 陳冠辰 (原<線段塗色問題>)



題目敘述

經過了一個月的對抗，雙方都搞得烏煙瘴氣的，於是 Player 和怪物們決定停止這場紛爭，握手言和。

怪物們被邀請到了 Player 家的別墅討論戰後事宜。然而，因為上次的偷襲未果，苦力怕家的阿帕打算惡整 Player。

阿帕把弟弟們騙到 Player 家門口，開始玩他們最喜歡的畫地盤遊戲。說是要玩遊戲，但實際上是讓弟弟們在路上亂撒火藥，但是施行到一半便被開會結束的一夥人發現了。苦力怕媽媽一邊向 Player 道歉，一邊責罵阿帕，要他把現場恢復原狀。

苦力怕家族的每個人其實都有些不同，最大的不同就是，每個人自己做出的火藥和別人都不一樣，如果隨便破壞或清除就會直接爆炸，所以每個人都有專屬的火藥破壞剪，能夠用來安全取走自己的火藥。

原本打算惡搞的阿帕臉色難看，他只記得一開始自己灑下的是哪些火藥，後來門口的樣子都忘了。而弟弟們只記得當時他們玩遊戲的過程。

畫地盤遊戲很簡單，就是從起點格(1 號格)開始跑，沿途每一格撒下自己的火藥，直到自己想停下來為止。

沒錯，此時就輪到聰明的你拯救大家。已知一開始的火藥分布情況，和後來的火藥變化，為了安全的清除所有火藥，請你告訴 Player 現在的火藥分布情況。

為了方便表示，對於同一個苦力怕撒下的火藥，都用相同編號表示。沒有人的編號和別人重複。

輸入說明

第一行有一個數字 N ，代表門口的寬度。

接下來有 N 個數字，代表一開始火藥的分布情況。

第三行有一個數字 Q ，代表玩遊戲的場數。

接下來 Q 行每行有兩個數字 B_i, C_i ，代表第 i 次遊戲中停下來的位置和該苦力怕的編號。

輸出說明

輸出 N 個數字，代表最後門口的樣子(火藥分布情況)。

子任務表

必定符合 $1 \leq B_i \leq N \leq 1,000,000$ ， $1 \leq Q \leq 1,000,000$ ， $0 \leq C_i \leq 10^9$ 。

編號	分數	子任務內容
1	10 %	$1 \leq N, Q \leq 5,000$
2	12 %	$B_{i-1} \leq B_i$
3	25 %	$1 \leq N, Q \leq 300,000$
4	18 %	$B_{i-1} \geq B_i$
5	35 %	題目範圍

範例

範例輸入	範例輸出
5 0 0 0 0 0 3 4 1 2 2 3 3	3 3 3 1 0
5 1 1 1 4 1 5 3 2 3 1 3 3 1 1 2 4	4 4 3 4 1

提示

246 三硝基甲苯，原名 **Trinitrotoluene**，是俗稱的 TNT。

第八題：下水道 *Clean*

題目格式：互動題

時間限制：3.0 秒

記憶體限制：768 MB

作者：台南一中 / 周暘典



題目敘述

順利清理完自家門口的 Player 鬆了一口氣，心想麻煩終於都結束了。不料怪物們前腳剛走，鎮長馬上就找上門了。

因為最近下水道系統出了些問題，他希望 Player 能夠幫他清理清理。被問到為什麼鎮民不自己弄的時候，鎮長只哼了一聲，說：「誰叫你是唯一『空得出手』的人」。Player 瞬間無語。沒辦法，只好由你來幫忙了。

鎮裡的下水道是一條長形水管，每隔 5 公尺設置一個排水溝。因為不知道是哪些地方堵住了，只好試著把堵塞的地方找出來了。

你可以對下水道進行測試，對第 i 條水管前端注水，然後到第 j 條水管末端觀察，如果從 i 到 j 的這整段水管都沒被堵住，水就能直接流通；反之則無法。然後把找到的堵塞物給清理掉。

請你幫助鎮民把下水道給整理乾淨。

(假設下水道長度為 N ，下水道總共有 M 個堵塞物。)

互動說明

無論 C/C++，請一律使用 C++14 上傳。

請注意：請勿任意進行輸入/輸出。任何輸入/輸出可能造成評分結果為 0 分。

本題競賽頁面中有兩個檔案 “grader.h” 及 “example.cpp”。

請在程式開頭加上 `#include "grader.h"`，測試時請將範例檔案置於程式檔同一個資料夾。

互動時，有四個函數可以使用：

<code>int Init(int *N, int *M);</code>	於程式開始時呼叫此函數以取得 N, M。未呼叫此函數，其他函數皆無功效。 行為：呼叫後會將 N, M 設定完成，並初始化互動。 回傳：-1 代表初始化錯誤(呼叫超過一次)，1 代表初始化成功。
<code>int Ask(int L, int R);</code>	詢問水管 [L, R] 之間有沒有堵塞物。必須先呼叫 Init，否則函數無效。 回傳：-1 代表未初始化，-2 代表詢問錯誤(L>R 或 L, R 不在 1~N 之間)。 1 代表詢問結果為「有」，2 代表詢問結果為「沒有」。
<code>int Clean(int i);</code>	清理第 i 條水管。必須先呼叫 Init，否則函數無效。 回傳：-1 代表未初始化，-2 代表操作錯誤(i 不在 1~N 之間)。 1 代表操作成功(無論是否清理正確的位置)。
<code>void Submit();</code>	提交目前結果進入評分，並結束程式。若未呼叫 Init 則 0 分。 行為：將目前操作的結果送交評分。呼叫此函數後程式將自動停止。 評分方式請見下方。

可以參照“互動範例”或者是“example.cpp”。

計分方式

假設你的程式在時限內正確的運行，且正確清空了 X 個，呼叫 Ask 和 Clean 總共 Y 次。

如果 $X=M, Y \leq 300+M$ ，且正確呼叫 Init，將會獲得該筆 AC，其餘狀況則該筆 0 分。

評分結果

如果你的程式成功清理所有阻塞物，你會收到 “AC: You used ?? to clean.”

如果你的程式並未清理所有阻塞物，你會收到 “WA: Still Mess!”

如果你的程式清理了所有阻塞物，但是次數過多，你會收到 “WA: You used ?? times.”

以上是正常的回應訊息，其他的訊息包括：

“WA: Interaction Error.”：請檢查自己是否使用到輸入/輸出，以及是否正確呼叫 Init。

“SE: Server Error.”：請檢查自己是否使用到輸入/輸出。

“WA: Submit Error.”：請檢查自己是否使用到輸入/輸出，以及是否正確呼叫 Submit。

其他訊息例如 “Execution timed out” 將會正常顯示。

互動範例 (example.cpp)

範例程式 (C / C++)

```
#include "grader.h"

int main()
{
    int N, M;
    Init(&N, &M);          // N, M will be set after calling this.
    int x = Ask(1, 1);      // Ask if [1, 1] has anything.
    If (x==1) Clean(1);     // If yes, clean it.
    Submit();               // Submit the answer.
}
```

子任務表

必定符合 $20 \leq N \leq 1,000,000$, $1 \leq M \leq 10$ 。

編號	分數	子任務內容
1	10 %	$N \leq 300$
2	17 %	$N \leq 1,000$
3	19 %	$N \leq 10,000$, $M \leq 2$
4	24 %	$N \leq 1,000,000$, $M = 1$
5	30 %	題目範圍

關於 grader.h

本檔案為測試標頭檔。本機測試的時候請手動輸入 N, M 以及整個原始序列(0 代表乾淨、1 代表髒亂)。在該檔案裡面有範例測資可以使用。

請注意，本檔案和評測端檔案並不相同。

範例測資 (同 grader.h)

範例 A	範例 B
5 3 0 1 0 1 1	5 2 1 0 0 0 1
範例 C	
30 10 0 0 1 0 1 1 0 1 0 1 0 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1	