

Absurdistanské kráľovstvo je zaujímavá krajina. Obsahuje len zopár (presne n , $3 \leq n \leq 40\,000$) miest v rovine a medzi nimi sú len veľké, dlhé a rovné púšte, v ktorých sa nenachádza vôbec nič. Keďže ďalšie púšte na okolí sú už Absurdistanu na nič, tak aj jeho hranice sú definované ako konvexný obal jeho miest. V Absurdistane sú všetky mestá očíslované číslami od 1 po n a **žiadne tri** sa nenachádzajú **na jednej priamke**.

Nedávno sa kráľ rozhodol, že v mestách, ktoré ležia na hranici, postaví hraničné priechody. Aby to vedel naplánovať, však potrebuje vedieť, koľko tých hraničných miest vlastne je. A keďže to nie je triviálna úloha, ponúkol dcéru a presne polovicu kráľovstva tomu, kto mu s tým pomôže. Švárnemu šuhajovi Maťovi sa kráľova dcéra veľmi páči a polovica kráľovstva by sa mu tiež hodila. Maťo sa teda dal na túto úlohu. Maťo, súc skúsený riešiteľ algoritmických olympiád sa teda zamyslel a povedal si: “Veď to je vlastne jednoduché. Stačí si zistiť súradnice miest a vypísať, koľko z nich je na konvexnom obale.” Také jednoduché to však tento raz nie je. V Absurdistane totiž ešte nemajú spravené mapy. Preto jediný, kto aspoň niečo tuší o polohách miest, sú obchodní cestujúci.

V Absurdistane nie sú cesty, a preto koče obchodných cestujúcich jazdia medzi mestami priamo. Väčšina cestujúcich má už dlhoročné skúsenosti a prebrádzili všetky mestá kade tade. Preto keď sa ich spýtate na trojicu miest, tak hneď vedia, že keď idú z prvého mesta cez druhé do tretieho, tak či v druhom odbočujú doľava alebo doprava. (Ekvivalente, ak im povieť tri mestá ktoré tvoria vrcholy trojuholníka, tak vám povieť, či ste ich povedali v smere hodinových ručičiek, alebo nie.) Obchodní cestujúci však nie sú zhovorčiví, a preto sa ich Maťo môže spýtať najviac 1 000 000 otázok.

Veďte sa do ťažkej úlohy šuhaja Mateja a pýtáním sa obchodných cestujúcich zistíte, koľko hraničných priechodov treba postaviť.

Komunikácia s hodnotiacim programom

Váš program by mal na komunikáciu používať knižnicu, ktorá mu umožňuje pýtať sa obchodných cestujúcich otázky a odpovedať kráľovi na otázku, koľko hraničných priechodov treba postaviť. Knižnica (`trilib.h` pre C a C++) vám na interakciu poskytuje nasledujúce funkcie:

- `int get_n();`
Vráti počet miest v Absurdistane
- `bool is_clockwise(int a, int b, int c);`
Vráti `true`, ak vrcholí trojuholníka a, b, c ($1 \leq a, b, c \leq n, a \neq b \neq c \neq a$) sú dané v poradí podľa smeru hodinových ručičiek a `false`, ak sú dané v protismere hodinových ručičiek.
- `void give_answer(int s);`
Funkcia, ktorú zavolá váš program práve raz. Na to aby ste získali body s musí byť počet miest na hraniciach Absurdistanu. Po zavolaní tejto funkcie sa už nesmiete pýtať žiadne ďalšie otázky a váš program by sa mal bezodkladne ukončiť.

V Java, môžete používať triedu `trilib`, ktorá implementuje metódy s rovnakými parametrami a výstupmi:

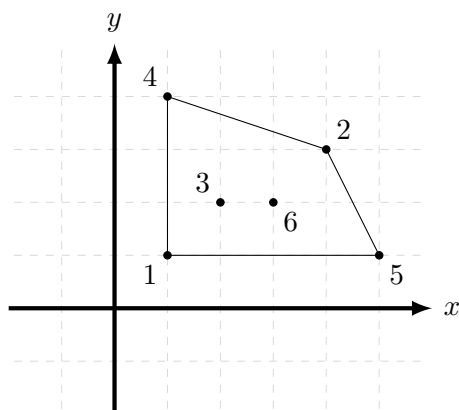
- `static public int get_n();`
- `static public boolean is_clockwise(int a, int b, int c);`
- `static public void give_answer(int s);`

V tejto úlohe je zakázané čítať zo štandardného vstupu alebo vypisovať na štandardný výstup.

Môžete predpokladať že polohy miest sú dané už pred spustením vášho programu. (Knižnica sa správa úplne deterministicky a nemení polohy na základe otázok, ktoré sa pýtate.) Preto napríklad v príklade nižšie môžete zavolať `give_answer(4)` bez pýtania sa otázok a test (vstup) bude vyhodnotený ako správny. Správnu odpoveď si teda môže váš program aj tipnúť bez toho, aby si bol úplne istý.

Príklad interakcie s hodnotiacim programom

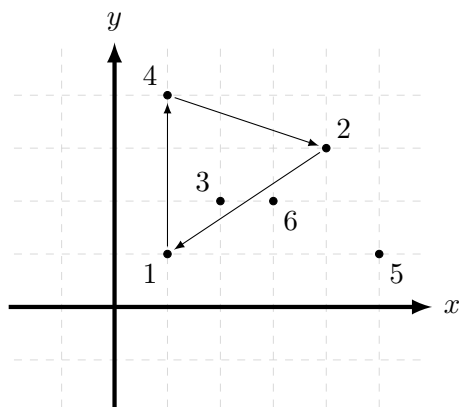
Predpokladajte $n = 6$ miest na súradniciach $(1, 1)$, $(4, 3)$, $(2, 2)$, $(1, 4)$, $(5, 1)$, $(3, 2)$ ako je nakreslené na obrázku nižšie. Konvexný obal (hranica Absurdistanu) je vyznačený čiarami. Obsahuje 4 vrcholy (mestá) na svojej hranici a preto treba 4 hraničné priechody.



Nasledujúca tabuľka ukazuje príklad interakcie s knižnicou, ktorý zodpovedá tomuto príkladu.

Volaná funkcia	Vrátená hodnota
<code>get_n()</code>	6
<code>is_clockwise(1, 4, 2)</code>	<code>true</code>
<code>is_clockwise(4, 2, 1)</code>	<code>true</code>
<code>is_clockwise(1, 2, 4)</code>	<code>false</code>
<code>is_clockwise(3, 6, 5)</code>	<code>true</code>
<code>give_answer(4)</code>	-

Obrázok zobrazuje trojuholník z prvej otázky. Mestá 1, 4, 2 sú v poradí podľa smeru hodinových ručičiek, takže vrátená hodnota je `true`.



Bodovanie

Testovacie dáta sú rozdelené do nasledujúcich podúloh s dodatočnými obmedzeniami. Vstupy v každej podúlohe sú rozdelené do jednej alebo viacerých testovacích skupín. Každá skupina obsahuje jeden alebo niekoľko vstupov. Body za skupinu dostanete, ak vyriešite všetky jej vstupy.

Vo všetkých vstupoch $3 \leq n \leq 40\,000$. Funkciu `is_clockwise` môžete zavolať najviac 1 000 000 krát.

Podúloha	Obmedzenia	Body
1	$n \leq 50$	15
2	$n \leq 500$	20
3	$n \leq 15\,000$	20
4	najviac jedno mesto nie je na hranici	20
5	žiadne ďalšie obmedzenia	25

Experimentovanie

V priečinku `public` vo vašom domovskom adresári je príklad knižnice, ktorá vám dovoľuje otestovať formálnu správnosť vášho riešenia. Knižnica načítava popis Absurdistanu zo štandardného vstupu v nasledujúcom formáte:

- v prvom riadku je prirodzené číslo n , počet miest,

- v ďalších n riadkoch: dve celé čísla, súradnice i -tého mesta.

Poskytnutá knižnica nerobí žiadne overenie správnosti vstupu a ani výstupu vášho riešenia. Na súťažnom servri je iná knižnica, ktorá dostáva korektné vstupy a kontroluje, či sa vaše riešenie **pýta korektné otázky** a či dáva správnu odpoveď.

Vstup z príkladu v tomto zadaní vo formáte vhodnom pre poskytnutú knižnicu sa nachádza v súbore `tri0.in`.

Po tom, čo je zavolaná funkcia `give_answer`, poskytnutá knižnica vypíše vašu odpoveď a počet otázok, ktoré ste sa pýtali na štandardný výstup.

Pre kompiláciu vášho riešenia s poskytnutou knižnicou môžete použiť nasledovné príkazy:

- C: `gcc -O2 -static trilib.c tri.c -lm -std=gnu99`
- C++: `g++ -O2 -static trilib.c tri.cpp -lm -std=c++11`

V Jave netreba použiť žiaden špeciálny program na kompiláciu.

Vaše riešenie by malo pri testovaní byť rovnakom priečinku ako poskytnutá knižnica.