



УПУТСТВО

Пред вама је 6 задатака чији је преглед дат у табели на следећој страни. **Ученици Б категорије решавају само прва 3 задатка а ученици А категорије само последња 3 задатка.** Уколико ученик који по пропозицијама припада Б категорији **преда бар један задатак из А категорије, сматра се да је одлучио да пређе у А категорију.** Задатке морате радити самостално и **коришћење интернета није дозвољено.** Израда задатака траје 5 сати.

Свој рад сачувајте на диску Z: **(не правите никакве фолдере/подфолдере, све програме чувате у кореном фолдеру).** Потребно је да сачувате искључиво **.pas, .c** или **.cpp** изворне (source) кодове **чија имена морају бити као у датој табели.** Оставите себи времена како бисте проверили да ли сте урадили све како треба.

Тестирање задатака се обавља под оперативним системом Linux, **на истој машини и систему као и за овогодишње квалификације,** уз коришћење званичних компајлера **FreePascal 2.4.4-3.1** и **GCC 4.6.3.** Приликом компајлирања, користе се следеће команде:

- За Pascal фајлове: `fpc -dEVAL -XS -O2 -o [ime_fajla] [ime_fajla].pas`
- За C фајлове: `gcc -DEVAL -static -O2 -o [ime_fajla] [ime_fajla].c -lm`
- За C++ фајлове: `g++ -DEVAL -static -O2 -o [ime_fajla] [ime_fajla].cpp`

Меморијска ограничења у задацима се односе на **укупну меморију** коју ваш програм користи у било ком тренутку – **HEAP + STACK** + величина самог програма. Осим ових ограничења, **не постоји** посебно ограничење за **STACK** меморију. Максимална дозвољена величина изворног кода је **100 KB.**

НАПОМЕНЕ

- Подаци се читају/исписују преко стандардног улаза и излаза - **немојте користити фајлове!**
- Излазни подаци морају бити **тачно** у облику датим у опису задатка. **Немојте исписивати додатне ствари** попут "Тражени број је...".
- На крају програма **обавезно уклонити** "readln;" и "system('pause');" наредбе!
- Уколико је потребно користити 64-битне бројеве, користите **int64** у Pascal-у, односно **long long** у C/C++-у; обратите пажњу да **long** у C/C++-у не мора увек бити 64-битни тип. Уколико за учитавање/испис 64-битних бројева у C/C++-у користите функције `scanf/printf`, потребно је употребити спецификатор `%lld`.
- Ознака за почетак нове линије може бити један карактер `'\n'` **ИЛИ** два карактера `"\r\n"`. Ваш програм мора исправно обрађивати **оба случаја** приликом учитавања података. Приликом учитавања препоручују се функције `read/readln` (Pascal) и `scanf` (C/C++).
- У C++ кодовима, користити `<iostream>` а не `<iostream.h>`. Такође, морате **експлицитно include-овати** све библиотеке чије функције користите (нпр. `<cstring>`, `<cstdlib>`, `<algorithm>`). У неким окружењима (DevC++) ваш код ће радити и без тога **али не и на званичном систему!**
- У C/C++ кодовима функција **main** мора бити декларисана као **"int main()"** а не као **"void main()"/"main()"**. Такође, ова функција мора враћати вредност, тј. морате имати **"return 0;"**.



ЗАДАТАК	АПСОЛУТНО БРОЈ	ИЗМЕНА	АНТИ-ЖМУРКЕ	ЗАМЕНА	БРОЈАЊЕ	КОНТРА- ЖМУРКЕ
категирија	Б	Б	Б	А	А	А
назив <i>source</i> кода	apsolutno.pas apsolutno.c apsolutno.cpp	izmena.pas izmena.c izmena.cpp	anti.pas anti.c anti.cpp	zamena.pas zamena.c zamena.cpp	brojanje.pas brojanje.c brojanje.cpp	kontra.pas kontra.c kontra.cpp
улаз	СТАНДАРДНИ УЛАЗ (stdin)					
излаз	СТАНДАРДНИ ИЗЛАЗ (stdout)					
временско ограничење	0.2 sec	1 sec	1 sec	0.5 sec	2 sec	1 sec
меморијско ограничење	64 MB	64 MB	64 MB	64 MB	128 MB	64 MB
број поена	100	100	100	100	100	100



Назив проблема: Апсолутно број

Временско ограничење: 0.2 секунде

Меморијско ограничење: 64 МБ

Текст проблема

Након целодневног рада на задацима и тест примерима, чланови Тајне комисије из сенке остају да поделе послове за сутра традиционалним алгоритмом.

Најпре запишу два низа A и B , сваки са N целих бројева. Након тога, N пута рачунају следећу суму $\sum_{i=1}^N |A_i - B_i|$ и након сваког рачунања ротирају низ A за једно место у десно. Све резултате на крају саберу и ово решење назову „сума свих сума“.

Чланови комисије користе „суму свих сума“ да одређеним алгоритмом поделе послове. Како рачунање ове суме одузима много времена, комисија је одлучила да напише програм. Али да би подела послова била фер, замолили су Вас да им помогнете и напишете програм који ће израчунати „суму свих сума“.

Улаз

У првом реду стандардног улаза налази се цео број N . У следећа два реда налази се по N целих бројева раздвојених размаком, низови A и B редом.

Излаз

У првом и једином реду стандардног излаза налази се један цео број, тражена „сума свих сума“.

Пример:

Улаз	Излаз
3 4 2 8 2 7 3	24

Објашњење примера

За почетни низ 4 2 8 сума разлика је $2 + 5 + 5 = 12$. Након ротирања низ A је 8 4 2, а сума разлика $6 + 3 + 1 = 10$. Након још једног ротирања, низ A је 2 8 4, а сума разлика $0 + 1 + 1 = 2$. После рачунања треће суме, ротирамо низ A али не рачунамо никакву суму. Тражена „сума свих сума“ је $12 + 10 + 2 = 24$.

Ограничења

- У 30% тест примера важи $1 \leq N \leq 10^3$ и $|A_i|, |B_i| \leq 10^3$
- У наредних 20% тест примера важи $1 \leq N \leq 10^5$ и $A_i, B_i \in \{0, 1\}$
- У преосталим тест примерима важи $1 \leq N \leq 10^5$ и $|A_i|, |B_i| \leq 10^6$



Назив проблема: Измена

Временско ограничење: 1 секунда
Меморијско ограничење: 64 МБ

Текст проблема

Мали Ђурица, бивши члан Тајне комисије из сенке, је купио робота. Ђурица је поставио робота на велику таблу (ради једноставности замислићемо да табла има бесконачну дужину и ширину) преко које је постављен координантни систем.

Уз робота је добио и низ команди Q , дужине N , који се састоји од слова L , R , D и U . Кад робот на позицији (x, y) учита слово q помера се на позицију (w, z) на следећи начин: ако $q = L$ онда $(w, z) = (x - 1, y)$, тј. иде лево; ако $q = R$ онда $(w, z) = (x + 1, y)$, тј. иде десно; ако $q = U$ онда $(w, z) = (x, y + 1)$, тј. иде горе; и ако $q = D$ онда $(w, z) = (x, y - 1)$, тј. иде доле. Поред овога, мали Ђурица је изабрао координату (r, s) .

Малог Ђурицу занима следеће:

- број A – колики је најмањи број команди које треба изменити у низу S тако да кад робот учита измењени низ и крене са позиције $(0, 0)$ у **неком** моменту прође кроз координату (r, s) ;
- број B – колики је најмањи команди које треба изменити у низу S тако да кад робот учита измењени низ и крене са позиције $(0, 0)$ **заврши** на координати (r, s) .

Помозите малом Ђурици и израчунајте A и B .

Улаз

У првом реду стандардног улаза налазе се три броја N , r и s раздвојена размаком. У другом реду се налази стринг Q дужине N .

Израз

У први и једини ред стандардног излаза исписати број A и број B раздвојени једним размаком.

Пример:

Улаз	Израз
6 0 2 RRRRRR	2 4
Улаз	Израз
13 3 -4 UURRRDUUDRLLL	3 4

Објашњење примера

У првом примеру ако робот прати низ UURRRR, који се разликује у 2 карактера од оног на улазу, наћи ће се на координати $(0, 2)$ после друге учитане команде. Ако робот прати низ команди LRLRUU наћи ће се на координати $(0, 2)$ након што учита цео низ.

У другом примеру могући низови команди који одговарају излазу су UDRRRDDDDRLLL и DURRRDDDDRLLL.



Ограничења

- $1 \leq N \leq 500\,000$.
- Q је састављен од карактера L , R , D и U , као што је описано у поставци проблема.
- $|r| + |s| > 0$.
- Низ ће увек моћи да се измени тако да бројеви A и B постоје.

Напомена

- Ако такмичар испише тачно број A добија 60% поена на датом тест примеру.
- Ако такмичар испише тачно број B добија 40% поена на датом тест примеру.

Тест примери су подељени у 4 дисјунктне групе:

- У тест примерима вредним 10 поена важи $s = 0$ и Q се састоји само од карактера L и R .
- У тест примерима вредним 10 поена важи $1 \leq N \leq 10$.
- У тест примерима вредним 30 поена важи $1 \leq N \leq 1\,000$.
- У тест примерима вредним 50 поена нема додатних ограничења.



Назив проблема: Анти-жмурке

Временско ограничење: 1 секунда
Меморијско ограничење: 64 MB

Текст проблема

Чланови Тајне комисије у сенци су прилично беспослени ових дана па у оближњем парку играју анти-жмурке. Анти-жмурке су потезна дечја игра у којој се један играч крије док га један или више играча, које називамо трагачима, тражи.

Парк је правоугаоних димензија и подељен је на јединичне квадрате. Свако квадратно поље представља жбун у којем се играч може сакрити. За сваког трагача је позната његова почетна позиција као и брзина која представља колико поља може прећи у једном потезу. Са тренутног поља трагач може прећи само на неко од четири суседна поља. Два поља су суседна уколико имају заједничку страну. Сви трагачи у исто време крећу са потрагом.

Ред је на малог Зокија да се сакрије. Он жели да буде што бољи у овој игри па жели да се сакрије у жбуну на пољу чија је **скривеност** највећа могућа. **Скривеност** неког поља је целобројна вредност која представља **у колико најмање потеза неки од трагача може стићи до тог поља крећући се најкраћим путем**. Помозите Зокију да пронађе поље са највећом скривеношћу јер су трагачи већ почели са одбројавањем.

Улаз

У првом реду стандардног улаза налазе се два природна броја N , M и K , који представљају димензије парка и број трагача, тим редом. У наредних K редова следе координате почетне позиције X_i и Y_i (редни број реда и колоне) и брзина V_i за сваког од трагача. Редови су нумерисани од 1 до N одозго надоле а колоне од 1 до M слева надесно.

Излаз

У првом и једином реду стандардног излаза исписати два броја i и j – редни број реда и колоне поља на којем се Зоки треба сакрити у складу са својим захтевом. Уколико постоји више решења исписати било које.

Пример:

Улаз	Излаз
5 7 2 1 2 3 4 4 1	3 7



Објашњење примера

Слика 1	Слика 2	Слика 3
1011122	6543456	1011122
1111222	5432345	1111222
1112223	4321234	1111223
2122233	3210123	2110123
2222333	4321234	2221233

Слике 1 и 2 представљају колико најмање потеза је потребно првом и другом трагачу да дођу до одговарајућих поља у парку, тим редом. На слици 3 видимо **скривеност** сваког поља. Приметимо да је највећа **скривеност** неког поља једнака 3 и да укупно таквих поља има 4. Било које од тих поља је решење.

Ограничења

- $1 \leq N, M, K \leq 1,000$
- $1 \leq V_i \leq 10^9$
- Никоја два трагача неће имати исте координате.

Напомена

Тест примери су подељени у 5 дисјунктних група:

- У тест примерима вредним 10 поена важи $K = 1$.
- У тест примерима вредним 20 поена важи $1 \leq N, M \leq 100$.
- У тест примерима вредним 20 поена важи да су брзине свих трагача **једнаке**.
- У тест примерима вредним 20 поена важи $1 \leq V_i \leq 50$.
- У тест примерима вредним 30 поена нема додатних ограничења.



Назив проблема: Замена

Временско ограничење: 0.5 секунди
Меморијско ограничење: 64 МБ

Текст проблема

Мали Пиксел, дугогодишњи члан Тајне Комисије у сенци, је одлучио да после много година одане службе Комисији поднесе оставку. Међутим, како не жели да његов одлазак протекне незапажено, одлучио је да шифрира свој последњи предлог задатка за Државно такмичење из програмирања, користећи кључ који је само њему познат. Осталим члановима Комисије оставио је само један низ који садржи све њихове идентификационе бројеве (**идентификациони бројеви су фиксне дужине и не морају бити јединствени**), као и текст проблема који чланови Комисије морају решити да би дошли до кључа:

Пиксел је најпре пролазио кроз низ, од првог елемента према последњем, све док није одабрао један идентификациони број и запамтио га. Затим је наставио пролаз кроз низ, и уколико би наишао на веома сличан идентификациони број (такав да се разликује од запамћеног **највише у једној цифри**) имао би *две опције*:

- да **запамти нови број уместо старог** и да настави даље кроз низ на исти начин;
- да **настави даље без промене запамћеног броја**.

Пиксел такође чува укупан број "бодова" који је освојио при пролазу кроз низ; у почетку број бодова је 0, међутим **при сваком памћењу новог броја, број бодова се повећава за апсолутну разлику цифре старог и новог броја на оној позицији у којој се нови број разликује од старог** (нпр. уколико је претходно запамћени број био 1234, а нови запамћени број је 1274, онда се број бодова повећава за 4). Потребно је одредити **највећи могући освојен број бодова**.

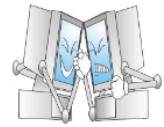
Државно такмичење из програмирања је већ почело, а Комисија није успела дешифровати задатак и није у могућности да такмичарима понуди нови. Стога су вас замолили за помоћ при дешифровању; као додатну мотивацију нуде вам 100 додатних бодова на Државном такмичењу.

Улаз

У првом реду стандардног улаза налази се природан број N , који представља укупан број чланова Тајне комисије у сенци. У сваком од наредних N редова налази се по један цео број A_i , који представља идентификациони број текућег члана Комисије у низу.

Излаз

У први и једини ред стандардног излаза исписати број M , који представља максималан број бодова који је Мали Пиксел могао да освоји.



Пример:

Улаз	Излаз
6 8823 2145 2185 3385 4145 4445	5

Објашњење примера

Пиксел најпре бира да запамти број 2145. Затим одмах има опцију да запамти 2185, међутим он одбија то да учини – пошто уколико одабере да запамти тек 4145, па 4445 недуго затим, осваја 5 бодова:

$$2145 \rightarrow 4145 \rightarrow 4445 \Rightarrow br_bodova = (4 - 2) + (4 - 1) = 5$$

Не постоји ни једна стратегија којом Пиксел може освојити више од 5 бодова.

Ограничења

- $1 \leq N \leq 10^5$
- Сви идентификациони бројеви A_i ће бити исте дужине.
- $0 \leq A_i < 10^7$

Напомена

Тест примери су подељени у три дисјунктне групе:

- У тест примерима вредним 30 поена важи $N \leq 1.000$.
- У тест примерима вредним 20 поена важи $A_i < 100$.
- У тест примерима вредним 50 поена нема додатних ограничења.



Назив проблема: Бројање

Временско ограничење: 2 секунде
Меморијско ограничење: 128 МБ

Текст проблема

Мали хакер Дрејко редовно покушава да избегне израду домаћих задатака. Како их није радио када је било време, а приближава се контролна вежба, схватио је да му се накупило N задатака које није урадио. Међутим, уместо да се истог тренутка баци на посао, уочио је много занимљиву тему за размишљање након што је сваком задатку доделио, по својој процени, неки природан број који представља колико је досадан (задаци којима су додељени већи бројеви су досаднији).

Дрејкова одлука је да реши неки **подниз узастопних** задатака из збирке. Међутим, тај подниз задатака ће одабрати тако да је најдосаднијем одабраном задатку придружен неки дати број k . Постоји потенцијално много оваквих избора задатака, а сваког иоле пристojног хакера занима **колико таквих избора постоји**. Пошто је ово мука и већег дела Тајне Комисије у Сенци, која броји Q чланова (при чему су сви сагласни са Дрејковом метриком досаде), поменућа Комисија ће вас богато наградити уколико јој решите проблем.

Улаз

У првом реду стандардног улаза се задаје природан број N . У наредном реду се задаје N природних бројева који представљају колико је одговарајући задатак у збирци досадан. У трећем реду се задаје природан број Q , након чега се у наредних Q редова налази по један природан број k , који представља упит "колико постоји поднизова узастопних задатака из збирке таквих да је вредност најдосаднијег задатка k ?".

Излаз

У Q редова стандардног излаза исписати одговоре на упите из улаза оним редом којим су били задати. Пошто су излазни бројеви потенцијално велики, исписати одговоре **по модулу** $10^9 + 7$.

Пример:

Улаз	Излаз
5	2
1 2 3 4 3	4
3	8
2	
3	
4	

Објашњење примера

Једини поднизови узастопних задатака чији је максимум досаде једнак 2 су [2] и [1,2].

Поднизови чији је максимум досаде једнак 3 су [1,2,3], [2,3], [3] и [3].

Поднизови чији је максимум досаде једнак 4 су [1,2,3,4], [1,2,3,4,3], [2,3,4], [2,3,4,3], [3,4], [3,4,3], [4], [4,3].



Ограничења

- $1 \leq N \leq 10^6$
- $1 \leq Q \leq 10^6$
- "Досадности" задатака су природни бројеви који нису већи од 10^9 .
- За све k из улаза важи $1 \leq k \leq 10^9$.

Напомене

Тест примери су подељени у дисјунктне групе.

- У тест примерима вредним 10 поена важе ограничења: $N \leq 10$, $Q \leq 10$, "досадности" задатака нису веће од 10.
- У тест примерима даље вредним 10 поена је задовољено $N \leq 1000$ и ниједном задатку неће бити придружена "досадност" већа од 1000.
- У тест примерима даље вредним 30 поена је задовољено $N \leq 100$.
- У преосталим тест примерима нема додатних ограничења.



Назив проблема: Контра-жмурке

Временско ограничење: 1 секунда
Меморијско ограничење: 64 MB

Текст проблема

Чланови Тајне комисије у сенци су прилично беспослени ових дана па у оближњем парку играју контра-жмурке. Контра-жмурке су потезна дечја игра у којој се један играч крије док га један или више играча, које називамо трагачима, тражи.

Парк је правоугаоних димензија и подељен је на јединичне квадрате. Свако квадратно поље представља или проходни жбун или непроходно дрво. За сваког трагача је позната његова почетна позиција као и брзина која представља колико поља може прећи у једном потезу. Са тренутног поља трагач може прећи само на неко од четири суседна поља. Два поља су суседна уколико имају заједничку страницу. Сви трагачи у исто време крећу са потрагом.

Ред је на малог Зокија да се сакрије. Он жели да буде што бољи у овој игри па жели да се сакрије у жбуну на пољу чија је **скривеност** највећа могућа. **Скривеност** неког поља је целобројна вредност која представља **у колико најмање потеза неки од трагача може стићи до тог поља крећући се најкраћим путем**. Помозите Зокију да пронађе поље са највећом скривеношћу јер су трагачи већ почели са одбројавањем.

Улаз

У првом реду стандардног улаза налазе се два природна броја N и M , који представљају димензије парка. Сваки од наредних N редова садржи низ од M цифара из скупа $\{0, 1\}$ **без размака** – изглед парка. 0 означава да се на одговарајућем месту налази жбун, а 1 дрво. Затим следи број трагача K и у наредних K редова координате почетне позиције X_i и Y_i (редни број реда и колоне) и брзина V_i за сваког од њих. Редови су нумерисани од 1 до N одозго надоле а колоне од 1 до M слева надесно.

Излаз

У првом и једином реду стандардног излаза исписати два броја i и j – редни број реда и колоне поља на којем се Зоки треба сакрити у складу са својим захтевом. Уколико постоји више решења исписати било које.

Пример:

Улаз	Излаз
5 7 0010000 0100000 0000110 0000000 0000000 2 1 2 3 4 4 1	2 6



Објашњење примера

Слика 1	Слика 2	Слика 3
10X3344	67X3456	10X3344
1X23334	5X32345	1X22334
1222XX4	4321XX4	1221XX4
2223334	3210123	2210123
2233344	4321234	2221234

На датим сликама непроходна поља означена су X-ом. Сlike 1 и 2 представљају колико најмање потеза је потребно првом и другом трагачу да дођу до одговарајућих поља у парку, тим редом. На слици 3 видимо **скривеност** сваког поља. Приметимо да је највећа **скривеност** неког поља једнака 4 и да укупно таквих поља има 5. Било које од тих поља је решење.

Ограничења

- $1 \leq N, M \leq 1,000$
- $1 \leq K \leq 1,000,000$
- $1 \leq V_i \leq 10^9$
- У парку ће бити проходног пута од сваког до сваког жбуна.
- Никоја два трагача неће имати исте координате.
- Почетне позиције трагача ће бити на пољу са жбуном.

Напомена

Тест примери су подељени у 4 дисјунктне групе:

- У тест примерима вредним 20 поена важи да су брзине свих трагача **једнаке**.
- У тест примерима вредним 20 поена важи $1 \leq N, M \leq 400$, $1 \leq V_i \leq 400$.
- У тест примерима вредним 20 поена важи $1 \leq K \leq 1,000$ и **сва поља су проходна**.
- У тест примерима вредним 40 поена нема додатних ограничења.