Проблем 1. Скупови

Дат је скуп A који садржи n различитих природних бројева. Одредити колико има подскупова скупа A за које важи да је збир највећег и најмањег елемента једнак датом броју m.

Улаз. (Улазни подаци се налазе у датотеци `skupovi.in`) У првом реду се налазе природни бројеви n и m ($2 \leqslant n \leqslant 100.000$, $1 \leqslant m \leqslant 1.000.000.000$). У другом реду се налази n различитих природних бројева, који представљају скуп A.

Излаз. (Излазне податке уписати у датотеку `skupovi.out`) У првом и јединoм реду треба исписати остатак при дељењу броја подскупова код којих је збир највећег и најмањег елемента једнак m и броја 1.000.000.007.

Пример 1.

skupovi.in	skupovi.out
5 9 7 2 9 5 4	5

Пример 2.

skupovi.in	skupovi.out
6 11 2 4 6 8 10 12	0

Објашњење. У првом примеру тражени подскупови су $\{2,7\}$, $\{2,4,7\}$, $\{2,5,7\}$, $\{2,4,5,7\}$ и $\{4,5\}$. Приметимо да скуп $\{9\}$ не задовољава услове задатка, пошто је збир највећег (број 9) и најмањег (број 9) елемента једнак 18. У другом примеру не постоји подскуп код кога је збир највећег и најмањег елемента непаран број, па је зато решење 0.

Проблем 2. Књижица

Кићи и Ћики су већ дуго најбољи другари. Нажалост, Кићи је изгубио памћење и Ћики је одлучио да му спреми специјално изненађење како би му помогао да се сети свега. Он је све њихове заједничке успомене записивао у једну зелену свескицу, и сада он жели да од тога направи једну од оних књижица где је читаоцу повремено понуђено да изабере како ће се прича наставити. На крају сваког дела приче, читалац бира да ли ће наставити онако како би Кићи наставио или онако како би Ћики наставио, с тим што се са неких делова не може наставити даље (на њима се прича завршава), а на појединим деловима постоји само једна могућност да се настави прича (некад је то Ћикијева а некад Кићијева идеја). У току једног читања не можемо два пута да се нађемо на истом делу и до сваког дела приче се може стићи на јединствен начин. Сваки део приче се налази на по једној страници.

Делови ове приче су нумерисани бројевима од 1 до n (где је n број делова приче) али пошто је Ћики перфекциониста он жели да их ренумерише (опет бројевима од 1 до n). Ћики је увек волео мале ствари, а Кићи велике, па зато Ћики жели да: ако са дела приче који је на страници a прелазимо на део приче који је на страници b важи:

- Ако смо изабрали оно што би Кићи урадио, онда је $b > a$, и редни број сваке странице до које ћемо касније у току читања приче моћи да дођемо је већи од a
- Ако смо изабрали оно што би Ћики урадио, онда је $b < a$, и редни број сваке странице до које ћемо касније у току читања приче моћи да дођемо је мањи од a

Помозите Ћикију да сложи странице како би што пре могао да да књижицу Кићију (Ћики је сјајан програмер, али је тренутно превише тужан да би размишљао о овом проблему).

Улаз. (Улазни подаци се налазе у датотеци `knjizica.in`) У првом реду налазе се три броја n, k и c ($1 \leqslant n \leqslant 250.000$, $1 \leqslant k, c \leqslant n$). Затим се у наредних k редова налазе по два броја, a и b, који означавају да са дела приче обележеног бројем a ако послушамо Кићија прелазимо на део обележен бројем b. Слично се у наредних c редова налазе по два броја, a и b, који означавају да са дела приче обележеног бројем a ако послушамо Ћикија прелазимо на део обележен бројем b. Пре ренумерисања почетак приче је на делу обележеном бројем 1 (обратите пажњу да после ренумерације не мора бити).

Излаз. (Излазне податке уписати у датотеку `knjizica.out`) Треба да се састоји од n редова - у i-том реду налази се број странице на којој ће се део приче обележен бројем i налазити после ренумерације. Ако постоји више решења, штампати било које.

Пример 1.

knjizica.in	knjizica.out
7 3 3	3
3 6	2
1 3	6
5 7	1
1 2	4
3 5	7
2 4	5

Проблем 3. Шифра

Драганче ради за контраобавештајну службу. Он је открио да се шифроване поруке непријатеља увек добијају линеарним пресликавањем. Најпре се свако слово представи бројем од 0 до $n-1$, где је n број слова у писму (ове бројеве ћемо звати индексима слова). Потом се шифра зада у облику $y = (a + b \times x) \mod n$, где је $\mathrm{nzd}(b, n) = 1$. То значи да се слово представљено бројем x шифрира словом које је представљено бројем y ($y = (a + b \times x) \mod n$). Поред сазнања о начину шифровања порука, Драганче има и податке о учестаности свих слова у језику непријатеља. За свако слово x се зна очекивани проценат појављивања $p(x)$ у произвољном тексту. Треба помоћи Драганчету да дешифрује једну изузетно важну поруку уз помоћ статистике и сазнања о линеарном шифровању.

Потребно је извршити дешифровање тако да се статистика у добијеном тексту што више поклапа са очекиваном статистиком. Квалитет поклапања за одређено слово се мери тако што се најпре израчуна учестаност тог слова у дешифрованом тексту $dc(x)$ (број појављивања слова x подељен са укупним бројем слова у тексту). Потом се квалитет поклапања $q(x)$ рачуна као $(p(x) - dc(x))^2$. Квалитет поклапања читавог дешифрованог текста се рачуна као сума квалитета поклапања свих слова.

Улаз. (Улазни подаци се налазе у датотеци `sifra.in`) У првом реду се налази природан број n, $3 \leqslant n \leqslant 26$, укупан број слова у непријатељском алфабету. Следи n редова и у сваком је дато једно слово из алфабета и један број, раздвојени размаком. Слово у првом реду има индекс 0, у другом 1, … у n-том реду има индекс $n-1$. Слово је увек представљено као велико слово из енглеског алфабета, док број представља проценат учестаности датог слова у произвољном тексту. Проценат је задат са две децимале. Слово имају различите учестаности, а укупан збир свих учестаности је 100. У следећем реду је цео број L ($10 \leqslant L \leqslant 1000$) и он представља дужину шифрованог текста. У последњем реду, налази се текст од тачно L слова (без размака и знакова интерпункције) који представљају шифровани текст. Сва слова која се налазе у шифрованом тексту су претходно већ задата са својим учестаностима.

Излаз. (Излазне податке уписати у датотеку `sifra.out`) У јединином реду излаза треба исписати дешифровани текст, дешифрован преко линеарног пресликавања, а који највише одговара задатој статистици.

Пример 1.

```
sifra.in
4
B 42.25
A 20.00
Z 10.15
E 27.60
10
AZABAZEBAZ
```

```
sifra.out
BEBABEZABE
```

Напомена. На основу улаза имамо да се слова представљају бројевима од 0 до 3 на следећи начин: $B \to 0$, $A \to 1$, $Z \to 2$ и $E \to 3$. Пресликавање код ког се добија најбоље преклапање је $y = (a + b \times x) \mod 4$, где су $a = 1$ и $b = 3$.

Проблем 4. Клизање

Учитељица Данка је одлучила да своје ђаке одведе на клизање. Међутим, пошто је једино она била расположена за такву авантуру, и ђаци осталих разреда су одлучили да се придруже. И тако је она са n ученика отишла на оближње клизалиште. Када су дошли, она је замолила сваког ђака да јој каже одакле ће почети да се клиза, и у ком правцу ће се клизати. Како је Данка веома паметна учитељица, она је на основу година сваког ђака и њихове снаге и особина закључила и којом брзином ће се свако клизати. Пошто веома дуго ради као учитељица, има довољно искуства да све ђаке може да контролише чак и ако се не помера (тј. све време стоји на само једном месту). Данка слабо види, а безбедност деце је на првом месту, па је одлучила да понесе наочаре. Али не било какве наочаре, већ специјалне наочаре на којима може да се намести колико далеко се види с њима (а може видети и лево и десно од своје позиције онолико далеко колико је подесила). Да би била сигурна да је све у реду, одлучила је да подеси даљину наочара тако да бар у једном моменту види бар k ученика. Пошто сте Ви њен најбољи ученик, замолила вас је да јој израчунате колика је најмања могућа даљина на коју треба да подеси наочаре и где треба да стоји тако да то важи.

Клизалиште је представљено једном бесконачном правом, позиције учитељице и ђака представљене су x-координатом на тој правој.

Улаз. (Улазни подаци се налазе у датотеци `klizanje.in`) У првом реду датотеке се налазе бројеви n ($4 \leqslant n \leqslant 1000$) и k ($1 \leqslant k \leqslant n$). У сваком од наредних n редова се налазе 2 броја: први број $x[i]$ ($0 \leqslant x[i] \leqslant 500.000$) означава почетну позицију ђака, а други број $v[i]$ ($-1000 \leqslant v[i] \leqslant 1000$) брзину ђака (односно колико ће се померити сваке секунде, ако је брзина негативна, то значи да се креће улево по x оси). Два или више ученика могу имати исту почетну позицију, али у том случају не могу имати једнаке брзине.

Напомена. Ђаци не мењају брзину током спуштања, и ниједан ђак неће "прегазити" учитељицу.

Излаз. (Излазне податке уписати у датотеку `klizanje.out`) У излазну датотеку треба исписати тражену најмању могућу даљину. Дозвољена релативна грешка је 10^{-8}. Ако је r коректно решење, а s решење које произведе ваш програм онда треба да важи:

$$\frac{2|r-s|}{|r|+|s|} \leqslant 10^{-8}$$

Пример 1.

klizanje.in	klizanje.out
`4 3`	`1.5`
`0 1`	
`2 4`	
`5 -2`	
`7 0`	

Објашњење. У тренутку 0.5, последња три клизача ће бити на позицијама $4, 4, 7$, редом па ће Данка, ако стоји на позицији 5.5 и подеси даљину на 1.5, видети њих тројицу.

Проблем 5. Вашари

Дошла је сезона вашара. Трговац Васа би да ове сезоне заради доста пара па хоће да све испланира унапред.

У више градова се организују вашари. Васа зна ког дана у неком граду почиње вашар и колико дана траје. Исто тако зна колика ће гужва бити на том вашару и колико може да заради у једном дану.

Али Васа не планира да продаје само у једном граду. Он планира да се сели из једног града у други и жели да му ми одредимо колико највише може да заради.

Улаз. (Улазни подаци се налазе у датотеци `vasari.in`) У првом реду се налазе два броја n и p ($1 \leqslant p \leqslant n \leqslant 500$). Број n представља број градова у којима се одржавају вашари, а p је редни број града у којем се Васа налази првог дана. У сваком од следећих n редова (сваки ред је везан за један град) се налазе по $n+3$ броја: a, b, c, $v[1], v[2], ..., v[n]$ ($1 \leqslant a, b \leqslant 200$, $1 \leqslant c \leqslant 1000$, $0 \leqslant v[i] \leqslant 400$). Број a представља ког дана у том граду почиње вашар, b представља колико дана траје, c представља колико Васа може да заради у једном дану на том вашару. Број $v[i]$ представља колико је потребно дана да Васа пређе из тог града у град i. Приметите да година може имати и више од 365 дана (али не више од 400 дана).

Излаз. (Излазне податке уписати у датотеку `vasari.out`) Треба да се састоји од само једног броја који представља колико Васа може да заради.

Пример 1.

vasari.in	vasari.out
3 2 5 6 5 0 3 3 1 5 2 2 0 1 7 8 3 1 2 0	37

Објашњење. Прва два дана Васа продаје у другом граду и заради 4. Онда два дана не продаје него се сели у први град. Тамо продаје током целог вашара и заради 30. Онда три дана прелази у трећи град и у њему продаје само један дан, и заради 3. Укупно је зарадио $4 + 30 + 3 = 37$.