

1. Feladat – biperm

100 pont

Egy n természetes szám ismeretében tekintsük az összes nem nulla, n -nél kisebb vagy egyenlő természetes szűmot, mindegyiket kétszer: $1, 1, 2, 2, 3, 3, \dots, n, n$. Ezeket a számokat tetszőlegesen összekeverjük és két sorba rakjuk, minden sorba n elemet. Egy ilyen adatstruktúrát bipermutációnak hívunk. Az 1., 2. és 3. ábrán egy-egy bipermutációt látunk $n=5$ -re.

Egy bipermutáció tökéletes, ha minden sorban egy-egy permutáció (lásd a 2. és 3. ábrákat).

Egy csere a p pozíión úgy van értelmezve, mint a p -ik oszlopban levő elemek felcserélése. Az alábbi példák esetében, a 2-ik, illetve a 3-ik ábrán levő tökéletes permutációkat az első ábrán levő bipermutációból nyertük miután egy műveletet alkalmaztunk a 2-ik pozíión, illetve 4 műveletet alkalmaztunk az 1, 2, 4 és 5-ik pozíciókon.

1	5	5	3	4
3	2	2	4	1

1. ábra

1	2	5	3	4
3	5	2	4	1

2. ábra

3	2	5	4	1
1	5	2	3	4

3. ábra

Követelmény

Ismerve egy bipermutációt, határozzátok meg:

- az egymástól különböző bipermutációk számát, amiket cserékkel tudunk létrehozni;
- a cserék minimális számát, melyek alkalmazása során tökéletes permutációt nyerünk;
- egy tökéletes bipermutációt, amelyet az eredetileg beolvastott bipermutációból nyerünk.

Bemeneti adatok

A biperm.in bemeneti állomány első sora tartalmazza n értékét. A következő két sor mindegyike n elemet tartalmaz szóközzel elválasztva, ami egy bipermutációt jelöl.

Kimeneti adatok

A biperm.out kimeneti állomány tartalma:

- az első sorban két természetes szám, szóközzel elválasztva. Az első szám az egymástól különböző bipermutációk száma, a második pedig a cserék minimális száma, amivel tökéletes permutációt nyerünk.
- a következő két sorban rendre n természetes szám egy-egy szóközzel elválasztva, amely egy tökéletes bipermutáció, amit az eredeti bipermutációból nyertünk.

Megszorítások és pontosítások

- $2 < n \leq 10\,000$;
- az egymástól különböző bipermutációk számának helyes kiszámítása 30%-ot ér az összpontból;
- a cserék minimális számának helyes kiszámítása 10%-ot ér az összpontból;
- egy tökéletes bipermutáció 60%-ot ér az összpontból. Több megoldás is létezhet, bármelyik helyes megoldás el van fogadva;
 - a különböző bipermutációk száma egyetlen teszt esetében sem haladja meg a 2 000 000 000-t.
 - egy helyes válasz csak akkor lesz elfogadva, ha az előtte levő válaszok léteznek, függetlenül ezek helyességétől;
 - a tesztek 40%-a esetében $n \leq 20$
 - a tesztek 40%-a esetében $20 < n \leq 400$
 - a tesztek 20%-a esetében $400 < n \leq 10\,000$

Példa

biperm.in	biperm.out	Magyarázatok																														
5 1 5 5 3 4 3 2 2 4 1	4 1 1 2 5 3 4 3 5 2 4 1	<p>Van 4 tökéletes permutációm. A cserék minimális száma 1 és két megoldás is létezik minimális cserével:</p> <table style="margin-left: 100px;"> <tr><td>1</td><td>2</td><td>5</td><td>3</td><td>4</td></tr> <tr><td>3</td><td>5</td><td>2</td><td>4</td><td>1</td></tr> </table> <p>és</p> <table style="margin-left: 100px;"> <tr><td>1</td><td>5</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>3</td><td>2</td><td>5</td><td>4</td><td>1</td></tr> </table> <p>A másik két megoldás, amit nem minimális cserével kapunk meg:</p> <table style="margin-left: 100px;"> <tr><td>3</td><td>2</td><td>5</td><td>4</td><td>1</td></tr> <tr><td>1</td><td>5</td><td>2</td><td>3</td><td>4</td></tr> </table> <p>A harmadik követelményt a fenti 4 megoldás bármelyike teljesíti.</p>	1	2	5	3	4	3	5	2	4	1	1	5	2	3	4	3	2	5	4	1	3	2	5	4	1	1	5	2	3	4
1	2	5	3	4																												
3	5	2	4	1																												
1	5	2	3	4																												
3	2	5	4	1																												
3	2	5	4	1																												
1	5	2	3	4																												

Maximális futási idő: 1 másodperc

Maximális memória: 128 MB amiből a verem mérete 64 MB

A forráskód maximális mérete: 10 KB

Problema 1 – biperm

100 puncte

Pentru un număr natural nenul n , să considerăm toate numerele naturale nenule mai mici sau egale cu n , luând fiecare număr de câte două ori: $1, 1, 2, 2, 3, 3, \dots, n, n$. Aceste numere le amestecăm aleator, și le aranjăm pe două linii a către n elemente. Structura astfel obținută o vom numi o *bipermutare*. În *figurile 1, 2 și 3* avem câte un exemplu de bipermutare pentru $n=5$.

O bipermutare este *perfectă*, dacă ambele linii ale structurii reprezintă câte o permutare (vezi *figurile 2 și 3*).

Prin *mutare pe poziția p*, înțelegem interschimbarea elementelor de pe aceeași coloană p . În exemplele de mai jos, bipermutarea perfectă din *figura 2* s-a obținut din bipermutarea din *figura 1*, aplicând o mutare pe poziția 2. Bipermutarea perfectă din *figura 3* s-a obținut din bipermutarea din *figura 1*, aplicând mutări pe pozițiile 1, 2, 4 și 5.

1	5	5	3	4
3	2	2	4	1

Figura 1

1	2	5	3	4
3	5	2	4	1

Figura 2

3	2	5	4	1
1	5	2	3	4

Figura 3

Cerințe

Cunoscând o bipermutare, determinați:

- numărul bipermutărilor perfecte distincte ce se pot obține prin mutări;
- numărul minim de mutări prin care se poate obține o bipermutare perfectă;
- o bipermutare perfectă obținută din bipermutarea inițială.

Date de intrare

Fișierul de intrare `biperm.in` conține pe prima linie valoarea lui n . Următoarele două linii conțin, fiecare, câte n elemente separate prin căte un spațiu, formând o bipermutare.

Date de ieșire

Fișierul de ieșire `biperm.out` va conține:

- pe prima linie două numere naturale separate printr-un spațiu, reprezentând numărul bipermutărilor perfecte distincte ce se pot obține din bipermutarea dată, respectiv numărul minim de mutări prin care se poate obține o bipermutare perfectă;
- pe următoarele două linii se vor tipări căte n numere separate prin spațiu, reprezentând o bipermutare perfectă obținută din bipermutarea dată.

Restricții și precizări

- $2 < n \leq 10\,000$;
- calculul corect al numărului bipermutărilor perfecte distincte valorează **30%** din punctaj;
- calculul corect al numărului minim de mutări valorează **10%** din punctaj;
- tipărirea unei bipermutări perfecte valorează **60%** din punctaj. Pot exista mai multe soluții, se va admite orice soluție corectă;
 - se garantează că numărul bipermutărilor perfecte distincte nu depășește **2 000 000 000** pentru niciun test.
 - acordarea punctajului la un răspuns corect este condiționată de existența răspunsurilor anterioare, indiferent de corectitudinea lor;
 - pentru **40%** din teste $n \leq 20$
 - pentru **40%** din teste $20 < n \leq 400$
 - pentru **20%** din teste $400 < n \leq 10\,000$

Exemplu

<code>biperm.in</code>	<code>biperm.out</code>	Explicații
5 1 5 5 3 4 3 2 2 4 1	4 1 1 2 5 3 4 3 5 2 4 1	Sunt 4 permutări perfecte. Numărul minim de mutări este 1 și există două soluții cu număr minim de mutări: $\begin{array}{ccccc} 1 & 2 & 5 & 3 & 4 \\ 3 & 5 & 2 & 4 & 1 \end{array}$ și $\begin{array}{ccccc} 1 & 5 & 2 & 3 & 4 \\ 3 & 2 & 5 & 4 & 1 \end{array}$ Celelalte două soluții, ce nu se obțin din număr minim de mutări sunt: $\begin{array}{ccccc} 3 & 2 & 5 & 4 & 1 \\ 1 & 5 & 2 & 3 & 4 \end{array}$ și $\begin{array}{ccccc} 3 & 5 & 2 & 4 & 1 \\ 1 & 2 & 5 & 3 & 4 \end{array}$ Pentru a treia cerință oricare dintre cele 4 soluții este acceptată.

Timp maxim de execuție per test: 1 secundă

Total memorie disponibilă: **128 MB** din care **64 MB** pentru stivă

Dimensiunea maximă a sursei : **10 KB**

2. feladat - subsecvente

100 pont

Legyen n egy természetes szám és $M = \{s_1, s_2, \dots, s_n\}$ egy nem üres, karaktersorokból álló halmaz, ahol minden karaktersorban az elemek az $\{ 'a', 'b' \}$ halmzból vannak véve. $|s_k|$ -val jelöljük az s_k sor karaktereinek számát, vagy másnépp megnevezve, az s_k hosszát. Az s_k egy részsora úgy van értelmezve mint az $i, i+1, \dots, j-1, j$ pozíciókon helyezkedő karakterek. Jelöljük ezt a részsort $s_k[i:j]$ -vel. Ezek szerint, ha $s_k = 'abbbaababa'$, akkor $s_k[3:6] = 'bbaa'$ vagyis a kiemelt részsor: $'abbbaababa'$.

Követelmény

Adott egy M halmaz, számítsuk ki egy részsor maximális hosszát, amelyik M minden karaktersorában megtalálható.

Bemeneti adatok

A subsecvente.in bemeneti állomány első sorában található egy n természetes szám, amely az M halmaz kardinális számával egyenlő. A következő n sor mindegyike az M halmaz egy-egy karaktersorát tartalmazza.

Kimeneti adatok

A subsecvente.out kimeneti állományba egyetlen számot kell beírni, a megtalált részsor hosszát.

Megszorítások

- $1 < n < 5$.
- Ha $|s| = |s_1| + |s_2| + \dots + |s_n|$, akkor $|s| < 50\ 001$.
- Garantáltan létezik minden esetben megoldás.
- A megoldás garantáltan nem nagyobb mint 60.
- A tesztek 30%-ra: $|s| < 101$.
- A tesztek 55%-ra: $|s| < 3\ 501$.
- A tesztek 80%-ra: $|s| < 10\ 001$.

Példa

subsecvente.in subsecvente.out Magyarázat

4
abbabaaaaabb
aaaababab
bbbbaaaab
aaaaaaaaabaaab

5

Egy maximális hosszúságú közös részsor hossza 5.

A mi példánkban az 5 hosszúságú közös részsor aaaab:
abbaba**aaaabb**, **aaaababab**,
bbbbaaaab, aaa**aaaabaaab**.

Maximális futási idő: 0.6s.

Maximális memória: 128MB amiből 64MB a verem mérete

A forráskód maximális mérete: 20KB

Problema 2 - subsecvențe

100 puncte

Fie n un număr natural și $M = \{s_1, s_2, \dots, s_n\}$ o mulțime de siruri de caractere nevide. Fie s_k un sir de caractere din M . Atunci, orice caracter al lui s_k aparține mulțimii { 'a', 'b' }. Notăm prin $|s_k|$ numărul caracterelor sirului s_k sau, echivalent, lungimea sa. O subsecvență $s_k[i:j]$ a lui s_k este formată din caracterele situate pe pozițiile consecutive $i, i+1, \dots, j$. Astfel, dacă $s_k = 'abbbaababa'$, atunci $s_k[3:6] = 'bbaa'$ sau subsecvența evidențiată: $'ab**bbaa**baba'$.

Cerință

Fiind dată o mulțime M , se cere să se determine lungimea maximă a unei subsecvențe care se găsește în toate sirurile din M .

Date de intrare

Pe prima linie a fișierului de intrare subsecvențe.in se găsește un număr natural n egal cu cardinalul mulțimii M . Pe fiecare din următoarele n linii se găsește câte un sir din mulțimea M .

Date de ieșire

Pe prima linie a fișierului de ieșire subsecvențe.out se va scrie un singur număr natural egal cu lungimea subsecvenței găsite.

Restricții

- $1 < n < 5$
- Dacă $|s| = |s_1| + |s_2| + \dots + |s_n|$, atunci $|s| < 50\ 001$
- Se garantează că va exista întotdeauna soluție
- Se garantează că rezultatul nu va depăși 60
- Pentru 30% din teste: $|s| < 101$
- Pentru 55% din teste: $|s| < 3\ 501$
- Pentru 80% din teste: $|s| < 10\ 001$

Exemplu

subsecvențe.in	subsecvențe.out	Explicație
4 abbabaaaaabb aaaababab bbbbaaaab aaaaaaaaabaaab	5	Lungimea unei subsecvențe comune de lungime maximă este 5. În exemplu subsecvența comună de lungime 5 este aaaab: abbaba aaaabb , aaaab abab, bbbbaaaab, aaaaaaab aaab.

Timp de execuție per test: 0.6s

Memorie per test: 128MB din care 64MB pentru stivă

Dimensiunea maximă a sursei: 20KB