



GeOlymp Series 2010

Finals

#	Problem Name	Time Limit	Memory Limit
A	standings	1 sec.	64 MB
B	lab	1 sec.	64 MB
C	compute	2 sec.	64 MB
D	boardwalk	2 sec.	64 MB
E	tournament	2 sec.	64 MB
F	triangle	1 sec.	64 MB
G	geolymp	1 sec.	64 MB
H	harmful	1 sec.	64 MB
I	permutation	1 sec.	64 MB

ამოცანა A. "შედგები"

ჯეოლიმპის ყოველი რაუნდის შემდეგ ჩვენ გვიწევს სისტემის მიერ დაგენერირებული საბოლოო შედეგების ორად გაყოფა, რათა შემოგთავაზოთ მოსწავლეების და სტუდენტების შედეგები ცალ-ცალკე. ზუსტად ამ ფუნქციის ავტომატიზაცია იქნება თქვენი დავალება.

შეზღუდვები:

შემომავალი კონტესტის შედეგი შედგება $1 \leq N \leq 100$ სტრიქონისაგან.

შემომავალი ფაილის ფორმატი:

standings.in ფაილის პირველ სტრიქონში მოცემულია N, მონაწილეთა რაოდენობა.

შემდეგ N სტრიქონში მოცემულია სისტემის მიერ დაგენერირებული კონტესტის შედეგები.

თითოეულ სტრიქონში მოცემულია თითო ჰარით გამოყოფილი მონაწილის პოზიცია, სახელი, გვარი და ფრჩხილებში ჩაწერილი სასწავლო დაწესებულება.

მონაწილის პოზიცია განისაზღვრება ერთი რიცხვით ან ტირით გამოყოფილი ორი რიცხვით და ბოლოვდება წერტილით. თუ მონაწილის პოზიციაა 3-7, ეს ნიშნავს რომ 5 მა მონაწილემ აჩვენა ერთიდაიგივე შედეგი და შესაბამისად დაიკავეს მე-3 დან მე-7-ს ჩათვლით ადგილები. მონაწილეები დალაგებული იქნებიან პოზიციის მიხედვით და პოზიციები სწორად იქნება მითითებული.

მონაწილის სახელი და გვარი წარმოადგენენ არაცარიელ დიდი და პატარა ლათინური ასოებისგან შემდგარ სიტყვებს.

სასწავლო დაწესებულება არის ორი ტიპის: უმაღლესი და სკოლა. სკოლის შემთხვევაში სასწავლო დაწესებულების დასახელება იწყება 'Sch' სიტყვით (ასოების რეგისტრი არსებითია, ანუ "SCH" სკოლის მოსწავლეს არ განსაზღვრავს). სასწავლო დაწესებულების სახელში სიტყვები ერთმანეთისაგან გამოყოფილია ერთი ჰარით. ყოველი სიტყვა შედგება დიდი და პატარა ლათინური ასოებისაგან, ციფრებისაგან და ', ' , ' ' , '#' სიმბოლოებისაგან.

თითოეული სტრიქონის სიგრძე არ აღემატება 100 სიმბოლოს.

გამომავალი ფაილის ფორმატი:

standings.out ფაილში დაბეჭდეთ ჯერ სტუდენტების შედეგები, შემდეგ ცარიელი ხაზი და შემდეგ მოსწავლეების შედეგები იგივე ფორმატით როგორც შემომავალ ფაილშია. საჭიროების შემთხვევაში, მოახდინეთ პოზიციების გადანომვრა. თუ რამოდენიმე მონაწილეს ერთიდაიგივე შედეგი აქვს, დაბეჭდეთ ისინი იმავე თანმიმდევრობით როგორც შემომავალ ფაილშია მოცემული.

შემომავალი ფაილის მაგალითი (standings.in)	გამომავალი ფაილის მაგალითი (standings.out)
8 1. Irakli Merabishvili (Spb SU) 2. Levan Varamashvili (Tbilisi SU) 3-5. Akaki Mamageishvili (Tbilisi SU) 3-5. Giorgi Nadiradze (Tbilisi SU) 3-5. Nodar Ambroladze (Sch #41) 6. Nika Gabisonia (Tbilisi SU) 7. Emzar Ulvashidze (Tbilisi SU) 8. Tsotne Tabidze (Sch #41)	1. Irakli Merabishvili (Spb SU) 2. Levan Varamashvili (Tbilisi SU) 3-4. Akaki Mamageishvili (Tbilisi SU) 3-4. Giorgi Nadiradze (Tbilisi SU) 5. Nika Gabisonia (Tbilisi SU) 6. Emzar Ulvashidze (Tbilisi SU) 1. Nodar Ambroladze (Sch #41) 2. Tsotne Tabidze (Sch #41)

ამოცანა B. "ლაბორატორია"

ერთ-ერთ ლაბორატორიაში იკვლევენ წყლის შემადგენლობას, რისთვისაც მეცნიერები იყენებენ N რაოდენობის წყლით სავსე ცილინდრს. ცილინდრების M რაოდენობის წყვილი შეერთებულია ერთმანეთთან მილით, რაც უზრუნველყოფს მათ შორის წყლის ცირკულაციას. ყოველი ცდის შემდეგ ცილინდრებიდან არათანაბრად ორთქლდება წყალი, ხოლო მომდევნო ცდის ჩატარებამდე აუცილებელია მათში წყლის სიმაღლეების გათანაბრება.

ყველა ცილინდრის ფუძის დიამეტრი ტოლია. ასევე მათ მთელ სიმაღლეზე მიჰყვებათ შკალა, რომლის საშუალებითაც შეიძლება განისაზღვროს თუ რა სიმაღლემდე ასხია მათში წყალი.

სამწუხაროდ, ცილინდრიდან წყლის გადასხმა შეუძლებელია, ხოლო მასში წყლის დონის ერთი ერთეულით ასამაღლებლად საჭიროა ელექტროენერგია. საჭირო ენერგიის მოცულობა დამოკიდებულია იმაზე, თუ რამდენ სხვა ცილინდრთან არის შეერთებული მოცემული ცილინდრი და ჯამში რა რაოდენობის წყალი ასხია მის მოსაზღვრე ცილინდრებში. კონკრეტულად, თუ A არის X ცილინდრის მოსაზღვრე ცილინდრებში წყლის სიმაღლეების ჯამის ხოლო B მოსაზღვრე ცილინდრების

რაოდენობა, X ცილინდრში წყლის დონის 1 ერთეულით ასაწევად საჭიროა $\frac{A}{B}$ ერთეული ელექტროენერგია.

მოცემული გაქვთ ცილინდრებში წყლის სიმაღლეები ცდის შემდეგ და ინფორმაცია მიღებით შეერთებულ წყვილებზე. თქვენი ამოცანაა დაადგინოთ ელექტროენერგიის საჭირო მინიმალური რაოდენობა, რომლის გამოყენებითაც შესაძლებელია ცილინდრებში წყლის სიმაღლეების გათანაბრება.

შეზღუდვები:

$$2 \leq N \leq 50$$

$$1 \leq M \leq 250$$

ყოველ ცილინდრში წყლის დონე მთელი რიცხვია $[0, 100]$ შუალედიდან.

ასევე გარანტირებულია, რომ წყალს ნებისმიერი ცილინდრიდან სხვა ნებისმიერ ცილინდრში შეუძლია მოხვედრა ცირკულაციის შედეგად.

შემომავალი ფაილის ფორმატი:

lab.in ფაილის პირველ სტრიქონში წერია ორი მთელი N და M რიცხვი. მომდევნო სტრიქონში მოცემულია ზუსტად N ნატურალური რიცხვი, რომელთაგან i-ური აღნიშნავს i-ურ ცილინდრში წყლის საწყის დონეს. მომდევნო M სტრიქონიდან თითოეული შეიცავს ორ ნატურალურ i და j რიცხვს, რაც აღნიშნავს რომ i-ური და j-ური ცილინდრები შეერთებულები არიან მილით.

გამომავალი ფაილის ფორმატი:

lab.out ფაილში დაბეჭდეთ წყლის დონეების გათანაბრებისთვის საჭირო მინიმალური ენერგია. თქვენი პასუხი სწორისგან არ უნდა განსხვავდებოდეს 0.01-ზე მეტით.

შემომავალი ფაილის მაგალითი (lab.in)	გამომავალი ფაილის მაგალითი (lab.out)
4 3 1 3 2 1 1 2 2 3 2 4	15.00

ამოცანა C. "გამოთვლა"

მოცემულია მიმდევრობა $A = \{A_1, A_2, \dots, A_N\}$. ყოველი i -სთვის 1-დან N -ის ჩათვლით, გამოთვალეთ $S_i = A_1 * \dots * A_{i-1} * A_{i+1} * \dots * A_N$. შემდეგ დაითვალეთ $S = S_1 + S_2 + \dots + S_N$ და გამოიტანეთ S -ის 1,000,000,009-ზე გაყოფისგან ნაშთი.

შეზღუდვები:

$$2 \leq N \leq 100,000$$

მიმდევრობის ყოველი წევრის მნიშვნელობა არის $[1, 1000000008]$ შუალედში.

შემომავალი ფაილის ფორმატი:

compute.in ფაილის პირველი სტრიქონი შეიცავს ერთ მთელ რიცხვს N . შემდეგ მოცემულია N რიცხვი, რომლებიც A მიმდევრობის წევრებს წარმოადგენენ.

გამომავალი ფაილის ფორმატი:

compute.out ფაილში გამოიტანეთ S -ის ნაშთი 1000000009-ზე გაყოფისგან.

შემომავალი ფაილის მაგალითი (compute.in)	გამომავალი ფაილის მაგალითი (compute.out)
3 2 4 5	38

განმარტება.

$$S_1 = A_2 * A_3 = 4 * 5 = 20$$

$$S_2 = A_1 * A_3 = 2 * 5 = 10$$

$$S_3 = A_1 * A_2 = 2 * 4 = 8$$

$$S = S_1 + S_2 + S_3 = 38$$

ნაშთი S -ის 1000000009-ზე გაყოფისგან 38-ის ტოლია.

ამოცანა D. "გასეირნება"

მოცემულია $N \times M$ უჯრედისგან შემდგარი დაფა. მის ზედა მარცხენა კუთხეში კოორდინატებით (1,1) დგას მანაო და უნდა ქვედა მარჯვენა (N,M) უჯრედში მოხვედრა. წერტილიდან მას მხოლოდ ქვევით ან მარჯვნივ შეუძლია მეზობელ უჯრედში გადასვლა, ანუ უჯრედიდან კოორდინატებით (i,j) იგი შეიძლება უჯრედებში (i+1,j) ან (i,j+1) გადავიდეს.

დაფაზე მონიშნულია K წერტილი, რომლებზეც მანაომ აუცილებლად უნდა გაიაროს. დაითვალებს, რამდენნაირად შეუძლია მას თავისი გზის გავლა. ვინაიდან ეს რიცხვი შეიძლება ძალიან დიდი იყოს, გამოიტანეთ მისი ნაშთი 1,000,000,009-ზე გაყოფისას.

შეზღუდვები:

$2 \leq N, M \leq 100,000$

$0 \leq K \leq 100,000$

K აუცილებლად მოსანახულებელი უჯრედის კოორდინატები [1..N], [1..M] შუალედებშია მოთავსებული.

შემომავალი ფაილის ფორმატი:

boardwalk.in ფაილის პირველი სტრიქონი შეიცავს სამ მთელ რიცხვს N, M და K.

მომდევნო K სტრიქონი შეიცავს თითოეული მონიშნული უჯრედის კოორდინატებს.

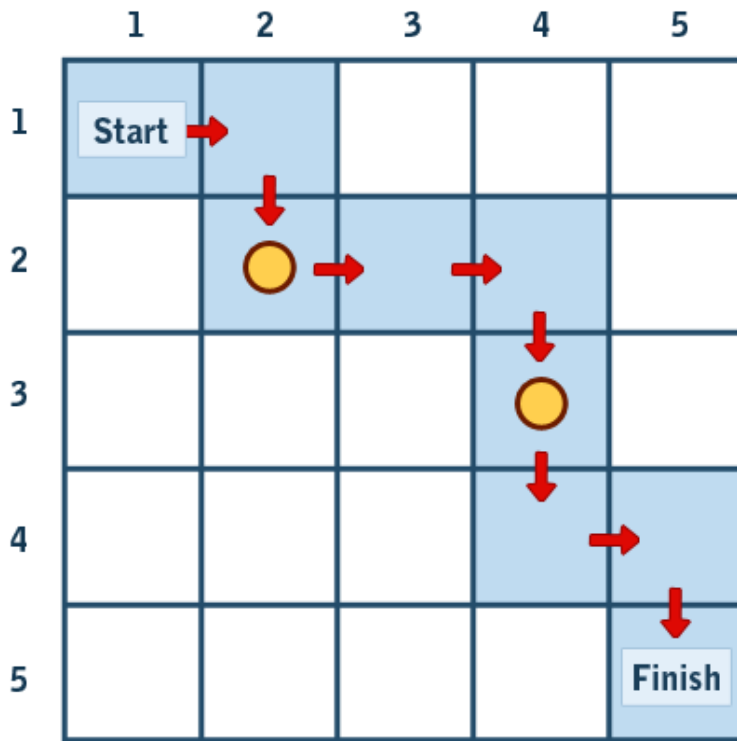
გამომავალი ფაილის ფორმატი:

boardwalk.out ფაილის ერთადერთ სტრიქონზე დაბეჭდეთ განხსვავებული გზების რაოდენობის ნაშთი 1000000009 გაყოფისგან.

შემომავალი ფაილის მაგალითი (boardwalk.in)	გამომავალი ფაილის მაგალითი (boardwalk.out)
2 2 2 3 4	18

განმარტება.

იხილეთ მანათს ერთ-ერთი შესაძლო მარშრუტი.



ამოცანა E. "ჩემპიონატი"

ჭ-ბურთის ჩემპიონატში N გუნდი იღებს მონაწილეობას. თითოეულ გუნდს თავისი მოედანი აქვს, რომელზეც მას ჩემპიონატის განმავლობაში ყველა სხვა გუნდთან მოუწევს თითო თამაშის ჩატარება. ანუ ჯამში ჩემპიონატი $N*(N-1)$ თამაშს მოიცავს. მატჩი ყოველთვის ერთ-ერთი გუნდის გამარჯვებით სრულდება და ამისთვის გამარჯვებული 1 ქულას იღებს.

ჩემპიონატის დასრულებისას 2 ცხრილს ადგენენ: Tournament Table და Rankings.

Tournament Table წარმოადგენს $N \times N$ ზომის მატრიცას, რომლის (i, j) ელემენტები (განსხვავებული i და j -სთვის) შეიცავენ i -ურ და j -ურ გუნდებს შორის i -ური გუნდის მოედანზე ჩატარებული მატჩის შედეგს.

Rankings წარმოადგენს ცხრილს, რომელშიც ჩემპიონატში მონაწილე ყველა გუნდი დალაგებულია დაგროვებული ქულების მიხედვით.

მაგალითად, განვიხილოთ შემდეგი Tournament Table:

	1	2	3	4
1		Win	Lose	Lose
2	Win		Lose	Win
3	Win	Lose		Win
4	Lose	Lose	Lose	

ცხრილი გვიჩვენებს, რომ გუნდმა №1 თავის მოედანზე მოუგო №2-ს და წააგო მესამესთან და მეოთხესთან. მეორე გუნდმა სახლში დაამარცხა გუნდები 1 და 4 და წააგო მესამესთან. მესამე გუნდმა თავის მოედანზე წააგო მეორესთან და მოუგო №1 და №4-ს. მეოთხე გუნდმა სახლში ყველა მატჩი წააგო.

შესაბამისად, ამ ჩემპიონატის Rankings ცხრილს შემდეგი სახე ექნება:

--	--

Team	Score
Team3	5
Team2	4
Team1	2
Team4	1

თუ ქულათა სხვაობა ჩემპიონატის ლიდერს და ბოლოადგილოსანს შორის აღემატება K-ს, მაშინ Rankings-ს *მეჩხერს* ეძახიან. დაითვალეთ, რამდენი ისეთი Tournament Table არსებობს, რომელსაც მეჩხერი Rankings ცხრილი შეესაბამება.

შეზღუდვები:

$2 \leq N \leq 6$

$0 \leq K \leq 9$

შემომავალი ფაილის ფორმატი:

tournament.in ფაილის ერთადერთი სტრიქონი შეიცავს რიცხვებს N და K.

გამომავალი ფაილის ფორმატი:

tournament.out ფაილში გამოიტანეთ ამოცანის პასუხი.

შემომავალი ფაილის მაგალითი (tournament.in)	გამომავალი ფაილის მაგალითი (tournament.out)
2 0	2
3 2	18

განმარტება.

პირველ მაგალითში შემდეგ ორ Tournament Table-ს შეესაბამება მეჩხერი Rankings:

	Lose
Win	

	Win
Lose	

ამოცანა F. "სამკუთხედი"

განვიხილოთ მართკუთხა სამკუთხედი. თვალსაჩინოა, რომ მართი კუთხის წვეროდან ჰიპოტენუზაზე სიმაღლეს თუ დავუშვებთ, ეს სამკუთხედი ორ მართკუთხა სამკუთხედად დაიყოფა.



ავიჩინოთ ამ სამკუთხედებიდან ერთ-ერთი და მეორე წავშალოთ. ჩვენ ისევ მართკუთხა სამკუთხედი გვექნება, რომელზეც იგივე ოპერაციის ჩატარება შეიძლება.



საწყისი სამკუთხედის კათეტების სიგრძეები თუ არის A და B , და მოცემული ოპერაცია K -ჯერ შევასრულებთ, რა იქნება დარჩენილი სამკუთხედის მაქსიმალური შესაძლო ფართობი?

შეზღუდვები:

$1 \leq A, B \leq 100$

A და B მთელი რიცხვებია.

$1 \leq K \leq 10$

შემომავალი ფაილის ფორმატი:

triangle.in ფაილის ერთადერთი სტრიქონი შეიცავს სამ მთელ რიცხვს A, B და K.

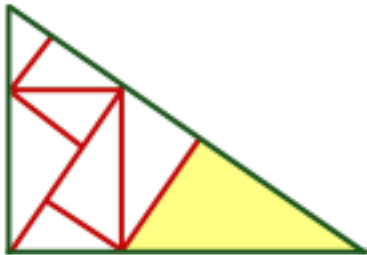
გამომავალი ფაილის ფორმატი:

triangle.out ფაილში გამოიტანეთ შემორჩენილი სამკუთხედის მაქსიმალური შესაძლო ფართობი. თქვენი პასუხი სწორისგან $1e-5$ -ზე მეტით არ უნდა განსხვავდებოდეს.

შემომავალი ფაილის მაგალითი (triangle.in)	გამომავალი ფაილის მაგალითი (triangle.out)
3 4 1	3.84
6 8 3	6.291456

განმარტება.

იხილეთ მეორე მაგალითის შესაბამისი ყველა შესაძლო დარჩენილი სამკუთხედი. მაქსიმალური ფართობის მქონე სამკუთხედი ყვითლად არის შეღებილი:



ამოცანა G. "GEOLYMP"

სტრიქონ **S**-ის ქვემიმევრობა ეწოდება ისეთ სტრიქონს, რომელიც მიიღება **S**-იდან 0 ან მეტი სიმბოლოს ამოშლით. მაგ. **ABC**, **AA** და **AD** წარმოადგენენ სტრიქონ **ABCD**-ს ქვემიმდევრობებს, ხოლო **ACB** და **XYZ** არა.

ჩვენ გვინტერესებს სტრიქონები, რომლებიც ქვემიმდევრობად სიტყვა **GEOLYMP**-ს შეიცავენ. კერძოდ, მოცემული **K**-სთვის უნდა იპოვოთ რომელიმე სტრიქონი, რომლის სიგრძე არ აღემატება 1000 სიმბოლოს და რომლის ზუსტად **K** ქვემიმდევრობა ტოლია **GEOLYMP**-ის.

შეზღუდვები:

$1 \leq K \leq 1,000,000,000$

შემომავალი ფაილის ფორმატი:

geolymp.in ფაილის ერთადერთი სტრიქონი შეიცავს ერთ მთელ რიცხვს **K**.

გამომავალი ფაილის ფორმატი:

geolymp.out ფაილში გამოიტანეთ არაუმეტეს 1000 ზედა რეგისტრის ლათინური ასოსგან შემდგარი სტრიქონი, რომლის ზუსტად **K** ქვემიმდევრობა არის **GEOLYMP**.

შემომავალი ფაილის მაგალითი (geolymp.in)	გამომავალი ფაილის მაგალითი (geolymp.out)
5	GXEOLYMPMPP

განმარტება.

მოცემული ტესტისთვის სწორი პასუხი ასევე იქნება **GABCDEOLYMPPPPP**, **GEOLYMPPPMP** და მრავალი სხვა.

ამოცანა H. "მავნე სუბიექტები"

მარჩიმ $N \times M$ უჯრედისგან შემდგარ დაფაზე დახატა ძალიან ლამაზი ფიგურა. კერძოდ, მან ზოგი უჯრედი შავად შეღება, ზოგი კი თეთრი დატოვა. ორ უჯრედს მეზობლები დავარქვათ, თუ მათ საერთო გვერდი გააჩნიათ. უჯრედების წყვილისთვის ვიტყვით, რომ ისინი ერთ ფიგურას ეკუთვნიან, თუ არსებობს უჯრედების ისეთი მიმდევრობა A_1, A_2, \dots, A_K , სადაც A_1 პირველი უჯრედი, A_K - მეორე და ყოველი i -ისთვის 2-დან K -ს ჩათვლით A_i და A_{i-1} უჯრედები მეზობლები არიან.

ამ დროს მოვიდნენ მავნე სუბიექტები სახელად ჟორა და მანაო. მათ გადაწყვიტეს დაფაზე დახატული ფიგურის გაფუჭება მისი 1 ან 2 შავი უჯრედის გათეთრებით. გაფუჭებად ისინი მოიაზრებენ ფიგურის დაშლას ორ ან მეტ ფიგურად, ანუ მათ მიერ უჯრედების გათეთრების შემდეგ დაფაზე 1-ზე მეტი ფიგურა უნდა აღმოჩნდეს.

უჯრედების გათეთრების ორი გეგმა ითვლება განსხვავებულად, თუ არსებობს ისეთი უჯრედი, რომელიც ერთი გეგმის შესრულების შემდეგ შავი რჩება, ხოლო მეორე გეგმის შესრულებისას თეთრდება. დაითვალეთ მავნე გეგმების საერთო რაოდენობა.

შეზღუდვები:

$1 \leq N, M \leq 50$

დაფაზე 2 მაინც შავი უჯრედი იქნება.

თავდაპირველად დაფაზე 1 ფიგურა ხატია, ანუ შავი უჯრედების ნებისმიერი წყვილი ერთ ფიგურას ეკუთვნის.

შემომავალი ფაილის ფორმატი:

harmful.in ფაილის პირველი სტრიქონი შეიცავს ორ რიცხვს N და M . შემდეგ არის N ხაზი, თითოეულ რომელთაგანზე M სიმბოლოა მოთავსებული – დაფის აღწერა. აქედან i -ური სტრიქონის j -ური სიმბოლო შეესაბამება დაფის (i, j) უჯრედს და იქნება '.', თუ ეს უჯრედი მარჩიმ შავად შეღება, ან 'X', თუ თეთრად დატოვა.

გამომავალი ფაილის ფორმატი:

harmful.out ფაილში გამოიტანეთ მავნე გეგმების რაოდენობა.

შემომავალი ფაილის მაგალითი (harmful.in)	გამომავალი ფაილის მაგალითი (harmful.out)
3 4 X.XX X.XX	10

განმარტება.

იხილეთ 10-ივე მავნე გეგმა:

X.XX	X.XX	X.XX	X.XX	XXXX
.X..	..X.	.XX.	.X.X	.X..
X.XX	X.XX	X.XX	X.XX	X.XX
X.XX	X.XX	X.XX	XXXX	X.XX
XX..	.X..	X.X.	..X.	..X.
X.XX	XXXX	X.XX	X.XX	XXXX

ამოცანა I. "გადანაცვლება"

ალბათ გაგინარდებათ, როდესაც გაიგებთ რომ ამ ამოცანას არანაირი ზღაპარი არა აქვს და პირდაპირ გთხოვთ რომ მოცემული N-სთვის დაბეჭდოთ 1-დან N-მდე რიცხვების ისეთი გადანაცვლება, რომ მის მეზობელ ელემენტებს შორის სხვაობის მინიმუმი იყოს მაქსიმალური შესაძლო 1..N რიცხვების ყველა გადანაცვლებას შორის.

შეზღუდვები:

$2 \leq N \leq 100$

შემომავალი ფაილის ფორმატი:

permutation.in ფაილის ერთადერთი სტრიქონი შეიცავს N-ს.

გამომავალი ფაილის ფორმატი:

permutation.out ფაილის პირველ სტრიქონზე გამოიტანეთ ისეთი მაქსიმალური შესაძლო K, რომლისთვისაც არსებობს 1..N რიცხვების გადანაცვლება, რომელშიც არც ერთ მეზობელ ელემენტთა წყვილში სხვაობა K-ზე ნაკლები არაა.

მეორე სტრიქონზე კი გამოიტანეთ შესაბამისი გადანაცვლება. თუ ასეთი გადანაცვლება რამდენიმეა, გამოიტანეთ ნებისმიერი.

შემომავალი ფაილის მაგალითი (permutation.in)	გამომავალი ფაილის მაგალითი (permutation.in)
4	2 3 1 4 2