NumLock

*Շարադրանք*

Կան ծրագրեր, որոնց միջոցով կարելի է վարժվել առանց ստեղնաշարին նայելու տեքստ հավաքելու գործում։ Երկու ամիս այդպիսի ծրագրով վարժվելուց հետո Արան բավականին արագացրել է տեքստ հավաքելու իր հմտությունը։

Վերջերս նա բավականին ծավալուն տեքստ հավաքեց իր նոութբուքով, բայց չէր նկատել, որ NumLock ստեղնը սեղմված է։ Արդյունքում, ամեն անգամ երբ նա սեղմում էր Home ստեղնը տողի սկիզբ գնալու համար, իրականում տպվում էր 7, իսկ երբ սեղմում էր End ստեղնը տողի վերջը գնալու համար՝ տպվում էր 1։

Պահանջվում է գրել ծրագիր տեքստը շտկելու համար։

*Մուտքային տվյալներ*

Մուտքում տրված է մինչև 100000 երկարությամբ մեկ տող։

*Ելքային տվյալներ*

Ելքում պետք է արտածել մեկ տող՝ շտկված տեքստը։

*Օրինակ*

|  |  |
| --- | --- |
| **Մուտք** | **Ելք** |
| First 7Second 1Third | Second First Third |

#### Փայտյա ձողիկներ

##### *Շարադրանք*

Դիցուք ունենք N հատ փայտյա ձողիկներ: Հայտնի են բոլոր ձողիկների երկարությունները և զանգվածները: Կտրող գործիքի միջոցով այդ ձողիկները պետք է հաջորդաբար մշակվեն: Ամեն ձողիկի մշակումից առաջ գործիքը նախապատրաստելու համար անհրաժեշտ է որոշ ժամանակ: Նախապատրաստման ժամանակը որոշվում է հետևյալ կերպ.

1. Առաջին ձողիկի մշակման համար նախապատրաստման ժամանակը 1 րոպե է:
2. l երկարությամբ և m զանգվածով ձողիկի մշակումից հետո անհրաժեշտ չէ նախապատրաստական աշխատանք եթե հաջորդը մշակվում է l՛ երկարությամբ և m՛ զանգվածով ձողիկը, որտեղ l ≤ l՛ և m ≤ m՛: Հակառակ դեպքում նախապատրաստման ժամանակը 1 րոպե է:

Անհրաժաշտ է գտնել բոլոր ձողիկները մշակելու համար նախապատրաստման ժամանակների մինիմալ գումարը:

Օրինակ, եթե ունենք 5 ձողիկ հետևյալ երկարություններով և զանգվածներով՝ (9,4), (2,5), (1,2), (5,3) և (4,1) ապա նախապատրաստման ժամանակների մինիմալ գումարը կլինի 2, քանի որ մշակումը կարող ենք իրականացնել հետևյալ հերթականությամբ` (4,1), (5,3), (9,4), (1,2), (2,5):

##### *Մուտքային տվյալներ*

Մուտքում տրված է երկու տող: Առաջին տողում տրված է N ձողիկների քանակը (1 ≤ N ≤ 5000): Հաջորդ տողում տրված են 2N հատ ամբողջ թվեր` ձողիկների երկարությունները և զանգվածները հաջորդաբար ll, m1, l2, m2, ... lN, mN:

##### *Ելքային տվյալներ*

Ելքում անհրաժեշտ է արտածել մեկ ամբողջ թիվ` նախապատրաստման ժամանակների մինիմալ գումարը:

##### *Օրինակ*

|  |  |
| --- | --- |
| **Մուտք** | **Ելք** |
| 5 9 4 2 5 1 2 5 3 4 1 | 2 |
| 3 2 2 1 1 2 2 | 1 |
| 3 1 3 2 2 3 1 | 3 |

Su-domino-ku

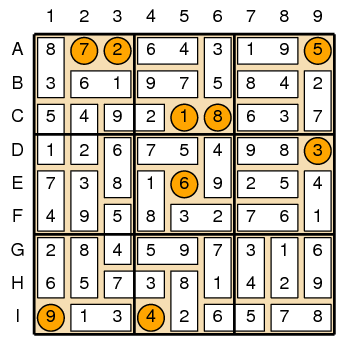
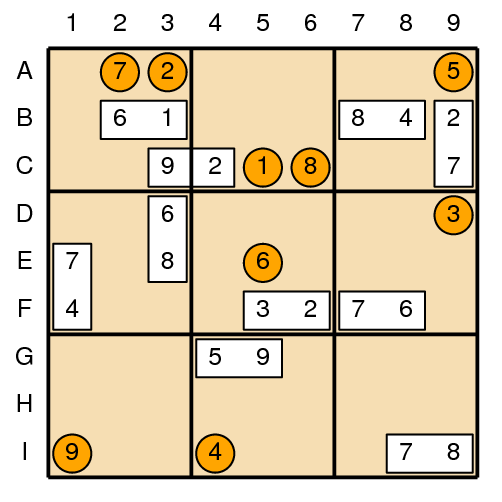
*Շարադրանք*

Սու-դոմինո-կու գլուխկոտրուկը սուդոկու խաղի տարբերակ է, որում դատարկ վանդակները պետք է լցնել դոմինոյի քարերի միջոցով։ Ստանդարտ սուդոկուի նման տրված է 9x9 չափի ցանց, որի վանդակները պետք է լցնել 1-ից 9-ը թվանշաններով։ Ճիշտ լրացնելու դեպքում

* Յուրաքանչյուր տող պետք է պարունակի 1-ից 9 բոլոր թվանշանները։
* Յուրաքանչյուր սյուն պետք է պարունակի 1-ից 9 բոլոր թվանշանները։
* Յուրաքանչյուր նշված 3x3 չափի քառակուսի պետք է պարունակի 1-ից 9 բոլոր թվանշանները։

Ինը կամայական վանդակներ սկզբնարժեքավորված են 1-ից 9 թվերով։ Մնում է 72 դատարկ վանդակ։ Դրանք պետք է լրացվեն դոմինոյի 36 քարերի միջոցով։ Դոմինոյի հավաքածուի մեջ 1-ից 9 թվերի յուրաքանչյուր թվազույգի համար (այսինքն՝ (1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (1,8), (1,9), (2,3), (2,4), (2,5), ...) կա մեկ խաղաքար։ Նկատենք՝ օրինակ (1,2)-ին և (2,1)-ին համապատասխանում է մեկ խաղաքար և, հետևաբար, այդ դոմինոն կարելի է պտտեցնել։ Նկատենք նաև, որ դոմինոները կարող են հատել 3x3 չափի քառակուսիեների եզրագծերը (ինչպես (2,9) դոմինոն ներքևի օրինակում)։

Օգնության կարգով ձեզ կտրվեն այնպիսի գլուխկոտրուկներ, որոնցում որոշ դոմինոներ տեղադրված են իրենց տեղերում։ Օրինակ՝ նկար 1-ում պատկերված է գլուխկոտրուկի սկզբնական դիրքը, նկար 2-ում պատկվերված է այդ գլուխկոտրուկի միակ լուծումը։



Նկար 1 Նկար 2

*Մուտքային տվյալներ*

Առաջին տողում տրված է սկզբնական դիրքում տեղադրված դոմինոների N քանակը (10 ≤ N ≤ 35): Հաջորդ N տողերից յուրաքանչյուրում նկարագրված է մեկ դոմինո a La b Lb քառյակի միջոցով։ a-ն դոմինոյի առաջին կեսի թիվն է, իսկ La-ն երկու սիմվոլանոց տող է, որը նկարագրում է a-ի դիրքը խաղատախտակի վրա՝ նկար 1,2-ում տրված կոորդինատային համակարգի միջոցով։ b թիվը և Lb տողը նկարագրում են դոմինոյի մյուս կեսի արժեքը և դիրքը։ Երաշխավորվում է, որLa-ն և Lb-ն նկարագրում են հարևան վանդակներ։

Մուտքի վերջին տողում տրված են 1-ից 9 թվերի դիրքերը՝ համաձայն վերը նկարագրվածի։ Մուտքում տրված բոլոր կոորդինատներն իրարից տարբեր են։

*Ելքային տվյալներ*

Ելքում պետք է արտածել գլուխկոտրուկի լուծումը 9x9 չափի թվանշանների աղյուսակի տեսքով, ինչպես ցույց է տրված օրինակում։

*Օրինակ*

|  |  |
| --- | --- |
| **Մուտք** | **Ելք** |
| 10 6 B2 1 B3 2 C4 9 C3 6 D3 8 E3 7 E1 4 F1 8 B7 4 B8 3 F5 2 F6 7 F7 6 F8 5 G4 9 G5 7 I8 8 I9 7 C9 2 B9 C5 A3 D9 I4 A9 E5 A2 C6 I1 | 872643195 361975842 549218637 126754983 738169254 495832761 284597316 657381429 913426578 |

Շարժական լաբիրինթ

*Շարադրանք*

Շատ դժվար ծրագրավորման խնդիրներ լուծելուց հետո, Կարենը որոշեց խաղալ գլուխկոտրուկ մի խաղ: Անհրաժեշտ է գրել ծրագիր, որը կօգնի Կարենին հեշտությամբ լուծել գլուխկոտրուկը:

Խաղը կոչվում է Շարժական Լաբիրինթ, այն խաղում են R տող և C սյուն ունեցող խաղատախտակի վրա: Խաղատախտակի ամեն վանդակի կենտրոնում կա սև կետ և հորիզոնական կամ ուղղահայաց գծեր դեպի տվյալ վանդակի հարևան վանդակները` հյուսիս, հարավ, արևելք, արևմուտք ուղղություններով: Հորիզոնական և ուղղահայաց գծերը կարող են բացակայել կամ միացված լինել հարևանների մի մասին: Ձեր առաջադրանքն է տեղափոխել խաղաքարը (i1, j1) վանդակի միջնակետից մինչև (i2, j2) վանդակի միջնակետ՝ շարժվելով միայն գոյություն ունեցող հորիզոնական և ուղղահայաց գծերով:

Անհրաժեշտ է հասնել վերջնակետին մինիմալ քայլերի միջոցով, որտեղ քայլը բաղկացած է հետևյալ երկու մասերից: Առաջին` ամեն քայլում կարելի է ընտրել որևէ վանդակ և այն պտտել 90 աստիճան ժամսլաքի կամ ժամսլաքի հակառակ ուղղությամբ: Երկրորդ` ամեն քայլում խաղաքարը կարելի է տեղափոխել ընթացիկ վանդակից հարևան վանդակ, շարժվելով միայն հորիզոնական և ուղղահայաց գծերով: Այսինքն A վանդակից կարելի է անցնել հարևան B վանդակը, եթե A-ն ունի սև գիծ միացված B-ին և հակառակը: Քայլի ընթացքում այդ երկու գործողություններից որևէ մեկը կարող է բացակայել:

*Մուտքային տվյալներ*

Մուտքի առաջին տողում գրված են R, C ամբողջ թվերը (1 ≤ R, C ≤ 20): Հաջորդ տողում գրված են i1, j1, i2, j2թվերը, հետևյալ սահմանափակումներով՝ 1 ≤ i1, i2 ≤ R և 1 ≤ j1, j2 ≤ C:

Հաջորդ R տողերից յուրաքանչյուրը պարունակում է ճիշտ C բառ, բաժանված բացակներով, ամեն վանդակին համապատասխանեցնելով մեկ բառ՝ ձախից աջ հերթականությամբ: Բառը ունի հետևյալ կառուցվածքը`

* Այն պարունակում է մեկ սիմվոլ՝ x, այս դեպքում վանդակը չունի հարևաններին միացնող գծեր:
* Կամ բառը պարունակում է N, E, S, W սիմվոլների մի մասը կամ բոլորը, որոնք համապատասպանաբար նշանակում են վանդակը միացված է գծով հյուսիս, արևելք, հարավ և արևմուտք:

Մուտքային օրինակները այնպիսին են, որ միշտ հնարավոր է (i1, j1)-ից հասնել (i2, j2) վանդակը:

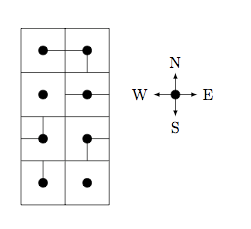
*Ելքային տվյալներ*

Ելքային ֆայլը պարունակում է մեկ ամբողջ թիվ` (i1, j1)-ից (i2, j2) վանդակը հասնելու համար անհրաժեշտ մինիմալ քայլերի քանակը:

*Օրինակ*

|  |  |
| --- | --- |
| **Մուտք** | **Ելք** |
| 4 2 1 1 4 1 E SW x EW NW ES N x | 5 |
| 2 1 1 1 2 1 E N | 1 |

Մուտքային խաղատախտակի տեսքը`



Առաջին օրինակում օգտագործվում են հետևյալ քայլերը՝

1. Պտտել (2,2)-ը ժամսլաքի ուղղությամբ: Տեղափոխվել (1,2) վանդակը:
2. Պտտել (3,2)-ը ժամսլաքի հակառակ ուղղությամբ: Տեղափոխվել (2,2) վանդակը:
3. Պտտել (3,2)-ը ժամսլաքի հակառակ ուղղությամբ: Տեղափոխվել (3,2) վանդակը:
4. Պտտել (3,1)-ը ժամսլաքի ուղղությամբ: Տեղափոխվել (3,1) վանդակը:
5. Պտտել (3,1)-ը ժամսլաքի ուղղությամբ: Տեղափոխվել (4,1) վանդակը: