

Descrierea problemelor – Ziua 2**Clasele 10 – 12**

Denumirea problemei	Numărul de puncte alocat problemei	Denumirea fișierului sursă
Papagalii	100	papagalii.pas papagalii.c papagalii.cpp
Poștașul	100	postas.pas postas.c postas.cpp
Șarpele	100	sarpe.pas sarpe.c sarpe.cpp
Senzori	100	senzori.pas senzori.c senzori.cpp
Total	400	-

Pentru a accesa Serverul ORI, introduceți în bara de adrese a navigatorului Internet (browser-ului):

ori.liceu:8080

Papagalii

Dan și Elena sunt pasionați de papagali. Pasiunile lor se realizează în mod diferit: fiind proprietarul unui magazin Zoo, Dan deschide o secție de vânzare a papagalilor, pe când Elena, odată cu deschiderea secției în cauză, începe să-și formeze la domiciliu o colecție proprie de papagali.

Inițial, în secția de vânzare nou deschisă papagali nu sunt. În continuare, procesul de interacțiune Magazin – Elena se desfășoară în felul următor:

1. În fiecare dimineață, Dan aduce la magazin S papagali.
2. Papagalii, aduși dimineața de Dan, imediat sunt puși în vânzare, împreună cu papagalii rămași din zilele precedente.
3. În fiecare zi, după pauza de masă, Elena vine la magazin și cumpără exact un papagal.

Evident, după N zile, Elena va avea acasă o colecție formată din N papagali.

Întrucât secția de vânzări abia a fost deschisă, pe parcursul tuturor celor N zile, unicul cumpărător de papagali a fost doar Elena.

Elena ar dori să știe, care este numărul C de colecții distincte ce pot fi formate din papagalii cumpărați din magazinul lui Dan, câte unul pe zi. Pentru Elena, ordinea în care papagalii ajung în colecția ei nu contează.

Sarcină. Elaborați un program, care, cunoscând numărul de papagali S și numărul de zile N , calculează numărul de colecții distincte C .

Date de intrare. Prima linie a intrării standard conține numerele întregi S și N , separate prin spațiu.

Date de ieșire. Ieșirea standard va conține pe o singură linie numărul întreg C .

Restricții. $1 \leq S \leq 10$; $1 \leq N \leq 40$. Timpul de execuție nu va depăși 0,05 secundă. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Fișierul sursă va avea denumirea `papagalii.pas`, `papagalii.c` sau `papagalii.cpp`.

Exemplul 1.

Intrare

2 2

Ieșire

5

Explicație. În scopuri didactice, vom nota cei doi papagali, aduși în prima zi, prin 1 și 2, iar cei doi papagali aduși în ziua a doua – prin 3, 4. Introducem în studiu mulțimea ordonată $\langle i, j \rangle$, unde i este numărul papagalului cumpărat în prima zi, iar j – numărul papagalului cumpărat în ziua a doua. Variantele posibile ale unor astfel de mulțimi sunt: $\langle 1, 2 \rangle$; $\langle 1, 3 \rangle$; $\langle 1, 4 \rangle$; $\langle 2, 1 \rangle$; $\langle 2, 3 \rangle$; $\langle 2, 4 \rangle$. Întrucât ordinea în care papagalii ajung în colecție nu contează, pentru Elena mulțimile ordonate $\langle 1, 2 \rangle$ și $\langle 2, 1 \rangle$ reprezintă una și aceeași colecție. Prin urmare, numărul de colecții distincte $C = 5$.

Exemplul 2.

Intrare

2 3

Ieșire

14

Postașul

Ion s-a angajat poștaș într-o localitate deluroasă. Chiar din prima zi el s-a convins că munca de poștaș este dificilă și responsabilă. Fiind pasionat la școală de informatică, el încearcă să-și raționalizeze munca.

În acest scop, Ion împarte localitatea în zone pătrate cu laturi de lungime egală și o reprezintă printr-un tablou Z cu n rânduri și n coloane. Elementele z_{ij} ale acestui tablou descriu zonele localității și pot avea următoarele valori caracteriale:

'P' – în zona în cauză se află oficiul poștal. Se consideră că localitatea are doar un singur oficiu poștal.

'K' – în zona în cauză se află o casă în care trebuie livrată trimiterea poștală.

'.' – zona în cauză reprezintă fie o porțiune de teren liber, fie o casă pentru care trimiteri poștale nu sunt (poștașul nu este obligat să viziteze astfel de case).

În fiecare zi, Ion vine la oficiul poștal, ia corespondența, o distribuie la casele din localitate și se întorce înapoi la oficiul poștal. În procesul de distribuire a poștei, Ion se deplasează din zona curentă în oricare din zonele adiacente, pe orizontală, pe verticală sau pe diagonală.

Acumulând o anumită experiență de muncă, Ion a observat că efortul depus de el în fiecare zi depinde de înălțimile dealurilor pe care trebuie să se urce. Pentru a ține cont de aceste înălțimi, Ion a alcătuit un tablou numeric H cu n rânduri și n coloane. Elementul h_{ij} al acestui tablou este egal cu altitudinea celui mai înalt din toate dealurile care se află în zona z_{ij} . Amintim, că altitudinea reprezintă înălțimea unui punct de pe suprafața pământului în raport cu nivelul mării.

Fiind un împătimit al științelor reale, Ion a intrudus în studiu o nouă mărime fizică – efortul zilnic E , necesar pentru distribuirea corespondenței. El a definit această mărime ca diferența dintre cea mai mare și cea mai mică din altitudinile zonelor pe care el le va parcurge în procesul de distribuire a corespondenței. Ion consideră că traseele pentru care mărimea E are valori mai mici, sunt mai bune.

Ajutați-l pe Ion să calculeze efortul minimal E_{min} cu care el ar putea să livreze corespondența zilnică.

Sarcină. Elaborați un program, care, cunoscând tablourile Z și H , calculează efortul E_{min} .

Date de intrare. Prima linie a intrării standard conține numărul întreg n . Fiecare din următoarele n linii ale intrării standard conține câte un șir din n caractere. Linia $(i + 1)$ a fișierului de intrare conține elementele rândului i al tabloului Z . În continuare, pe următoarele n linii urmează elementele tabloului H . Fiecare din aceste linii conține n numere întregi, separate prin spațiu. Linia $(i + 1 + n)$ a intrării standard conține elementele rândului i ale tabloului H .

Date de ieșire. Ieșirea standard va conține pe o singură linie numărul întreg E_{min} .

Restricții. $2 \leq n \leq 50$; $0 \leq h_{ij} \leq 10^6$. Timpul de execuție nu va depăși 2 secunde. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Fișierul sursă va avea denumirea `postas.pas`, `postas.c` sau `postas.cpp`.

Exemplul 1.*Intrare*

2
P.
.K
2 1
3 2

Ieșire

0

Explicație. Ion pornește din zona z_{11} (oficiul poștal), merge direct pe diagonală în zona adiacentă z_{22} (casa în care se livrează corespondența) și se întoarce înapoi în zona z_{11} . Efortul minim:

$$E_{min} = \max(h_{11}, h_{22}, h_{11}) - \min(h_{11}, h_{22}, h_{11}) = \max(2, 2) - \min(2, 2) = 2 - 2 = 0.$$

Exemplul 2.*Intrare*

3
P..
.KK
...
3 2 4
7 4 2
2 3 1

Ieșire

2

Explicație. Ion trece consecutiv următoarele zone: $z_{11}, z_{22}, z_{23}, z_{12}, z_{11}$. Efortul minim:

$$\begin{aligned} E_{min} &= \max(h_{11}, h_{22}, h_{23}, h_{12}, h_{11}) - \min(h_{11}, h_{22}, h_{23}, h_{12}, h_{11}) = \\ &= \max(3, 4, 2, 2, 3) - \min(3, 4, 2, 2, 3) = 4 - 2 = 2. \end{aligned}$$

Exemplul 3.*Intrare*

3
K.P
...
K.K
3 3 4
9 5 9
8 3 7

Ieșire

5

Explicație. Ion trece consecutiv următoarele zone: $z_{13}, z_{22}, z_{31}, z_{32}, z_{33}, z_{22}, z_{11}, z_{12}, z_{13}$. Efortul minim:

$$\begin{aligned} E_{min} &= \max(h_{13}, h_{22}, h_{31}, h_{32}, h_{33}, h_{22}, h_{11}, h_{12}, h_{13}) - \\ &\quad - \min(h_{13}, h_{22}, h_{31}, h_{32}, h_{33}, h_{22}, h_{11}, h_{12}, h_{13}) = \\ &= \max(4, 5, 8, 3, 7, 5, 3, 3, 4) - \min(4, 5, 8, 3, 7, 5, 3, 3, 4) = 8 - 3 = 5. \end{aligned}$$

Șarpele

Intr-o versiune simplificată a binecunoscutul joc clasic de calculator *Snake* (Șarpele), un șarpe trebuie să mănânce toate merele dintr-o livada. Regulile acestui joc sunt foarte simple:

1. În livada sunt împrăștiate pe pământ n mere, poziția fiecărui mar i fiind definită prin coordonatele carteziane întregi (x_i, y_i) .
2. Originea sistemului de coordonate se află în colțul stânga-jos al livezii. Axa de coordonate OX este orientată de la stânga la dreapta, iar axa OY – de jos în sus.
3. La începutul jocului, în livada intra un șarpe, care are drept scop să mănânce toate merele din livada.
4. Poziția curentă a șarpelui este definită prin coordonatele carteziane întregi (x_s, y_s) .
5. Poziția inițială a șarpelui este $(1, 1)$.
6. Repertoriul de comenzi ale șarpelui include instrucțiunile de deplasare SUS, JOS, STÂNGA, DREAPTA. Execuția unei astfel de comenzi constă în deplasarea șarpelui în direcția respectivă exact cu o unitate de lungime.
7. Atunci când coordonatele șarpelui devin egale cu coordonatele unui măr, șarpele mănâncă mărul respectiv.

Scopul jocului constă în deplasarea șarpelui în așa mod, încât toate merele să fie mâncate, iar drumul parcurs de șarpe să fie cât mai scurt.

Dorin este împătimit de jocul *Snake*. El îl joacă de multă vreme și a stabilit mai multe recorduri. Din păcate, pe consola lui de jocuri s-a defectat butonul comenzii JOS. În consecință, șarpele, indiferent de poziția în care se afla, poate executa doar comenzile SUS, STÂNGA, DREAPTA.

Dorin însă nu s-a descurajat și consideră această defecțiune o nouă provocare. Întrucât Dorin a devenit foarte priceput la acest joc, el dorește să stabilească un nou record. El nu se îndoiește deloc de abilitățile lui de a direcționa șarpele pe orice drum posibil, dar vrea să știe lungimea celui mai scurt drum, deplasându-se pe care șarpele ar mânca toate merele.

Sarcină. Elaborați un program, care, cunoscând coordonatele celor n mere, calculează lungimea L a celui mai scurt drum, deplasându-se pe care șarpele ar mânca toate merele.

Date de intrare. Intrarea standard conține pe prima linie numărul întreg n . Fiecare din următoarele n linii ale intrării standard conține numerele întregi x_i, y_i , separate prin spațiu. Linia $i + 1$ a intrării standard conține coordonatele mărului i .

Date de ieșire. Ieșirea standard va conține pe o singură linie numărul întreg L .

Restricții. $1 \leq n \leq 10000$; $1 \leq x_i, y_i \leq 10000$. Timpul de execuție nu va depăși 0,05 secunde. Programul va folosi cel mult 1 Megaoctet de memorie operativă. Fișierul sursă va avea denumirea `sarpe.pas`, `sarpe.c` sau `sarpe.cpp`.

Exemplu.

Intrare

5
2 2
5 3
7 3
8 4
4 6

Ieșire

16

Senzori de umiditate

Parcul Central din capitală este dotat cu un sistem de stropire cu comandă numerică. Stropitoarele din acest sistem sunt pornite și, respectiv, oprite, în funcție de umiditatea din zonele în care ele se află.

Pentru a măsura umiditatea, în zonele respective au fost montați K senzori. În scopuri didactice, senzorii sunt numerotați prin 1, 2, 3, ..., K .

Senzorii de umiditate comunică cu calculatorul ce comandă sistemul de stropire printr-o rețea Wi-Fi. Această rețea constă dintr-un punct de acces prin conexiune fără fir și N repetitoare. Raza de acțiune a punctului de acces este notată prin R . Repetitoarele sunt numerotate prin 1, 2, 3, ..., N , iar raza de acțiune a repetitorului i – prin r_i .

Amintim, că repetitoarele sunt destinate pentru extinderea ariei de acoperire (vezi *Figura 1*). Prin urmare, senzorii ce se află în raza de acțiune R a punctului de acces, comunică direct cu el, iar cei ce se află mai departe – prin intermediul unui canal de comunicații, format din unul sau mai multe repetitoare. Evident, pentru a forma un canal de comunicații, distanța d_{ij} dintre oricare două repetitoare vecine i, j din cadrul acestuia nu trebuie să depășească valoarea de $r_i + r_j$.

De exemplu, senzorul cu coordonatele (2, 18), marcat pe *Figura 1* cu o săgeată, poate comunica cu punctul de acces prin intermediul canalului de comunicație format din repetitoarele 1, 2, 4, 6 sau prin canalul format din repetitoarele 1, 2, 3, 5.

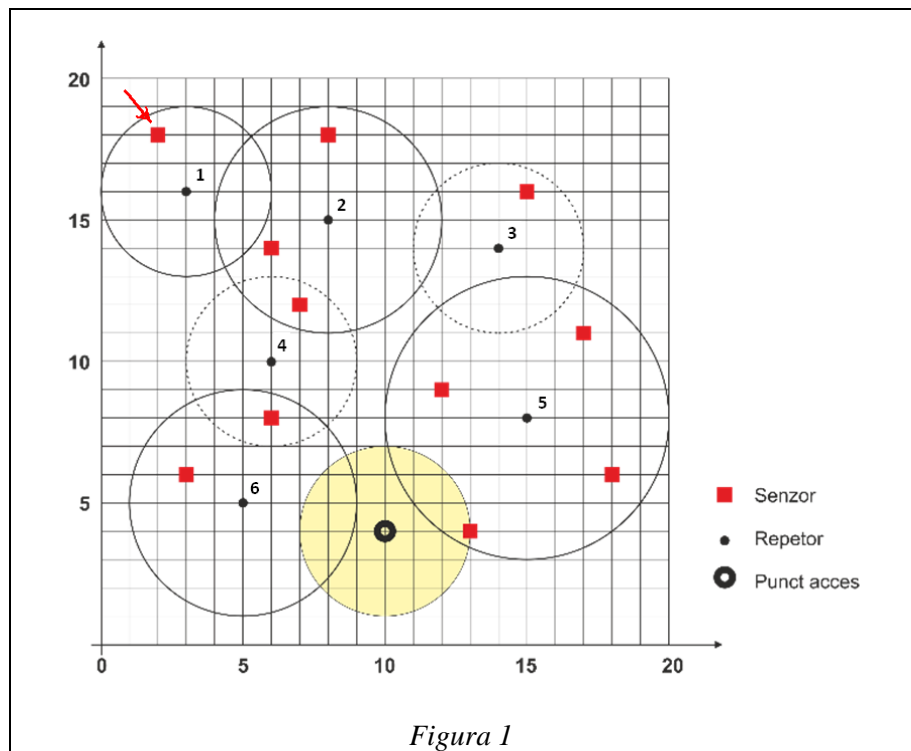


Figura 1

Cu timpul, din cauza condițiilor nefavorabile de exploatare, unele repetitoare ies din funcție. Întrucât sunt scumpe, administrația parcului a decis să înlocuiască nu repetitoare, ci acei senzori, care, din cauza defectării unor repetitoare, au pierdut legătura cu punctul de acces. Se preconizează că senzorii în cauză vor fi înlocuiți cu alții, de tip nou, care vor comunica cu calculatorul de comandă în mod direct, fără a mai folosi punctul de acces.

Pentru a-și planifica bugetul, administrația parcului dorește să cunoască numărul senzorilor ce vor trebui înlocuiți. Administrația are la dispoziție următoarea informație despre sistemul de stropire:

- coordonatele X, Y și raza de acțiune R a punctului de acces prin conexiune fără fir;
- numărul de repetitoare N , coordonatele x_i, y_i și raza de acțiune r_i a fiecărui repetitor i ;
- numărul de senzori K și coordonatele x_i, y_i ale fiecărui senzor;
- numărul de repetitoare stricate M și repetitoarelor propriu-zise, specificate prin numerele lor.

Sarcină. Elaborați un program, care, cunoscând informația despre sistemul de stropire, calculează numărul de senzori S ce trebuie înlocuiți.

Date de intrare. Prima linie a intrării standard conține descrierea punctului de acces prin conexiune fără fir: numerele întregi X, Y, R , separate prin spațiu.

Linia a doua a intrării standard conține numărul întreg N . Următoarele N linii ale intrării standard conțin descrierile repetoarelor. Linia $(i + 2)$ a intrării standard conține descrierea repetoarelor i : numerele întregi x_i, y_i, r_i , separate prin spațiu.

Următoarea linie a intrării standard conține numărul întreg K . Următoarele K linii ale intrării standard conțin descrierile senzorilor. Linia $(i + 2 + N + 1)$ a intrării standard conține descrierea senzorului i : numerele întregi x_i, y_i , separate prin spațiu.

Următoarea linie a intrării standard conține numărul întreg M . Următoarea, ultima linie a intrării standard conține M numere întregi separate prin spațiu: numerele repetoarelor stricate.

Date de ieșire. Ieșirea standard va conține pe o singură linie numărul întreg S .

Restricții. $1 \leq N \leq 1000$; $0 \leq M \leq N$; $1 \leq K \leq 1000$. Raza de acțiune R a punctului de acces, razele de acțiune r_i ale repetoarelor, $i = 1, 2, 3, \dots, N$, sunt numere naturale din intervalul $[0, 50]$. Coordonatele X, Y ale punctului de acces, coordonatele x_i, y_i ale repetoarelor și coordonatele x_j, y_j ale senzorilor sunt numere întregi intervalul $[-2000, +2000]$. Timpul de execuție nu va depăși 0,1 secundă. Programul va folosi cel mult 8 Megaocteți de memorie operativă. Fișierul sursă va avea denumirea `senzori.pas`, `senzori.c` sau `senzori.cpp`.

Exemplu.

Intrare

10	4	3
6		
3	16	3
8	15	4
14	14	3
6	10	3
15	8	5
5	5	4
11		
2	18	
8	18	
15	16	
6	14	
7	12	
17	11	
12	9	
6	8	
3	6	
18	6	
13	4	
2		
3	4	

Ieșire

5

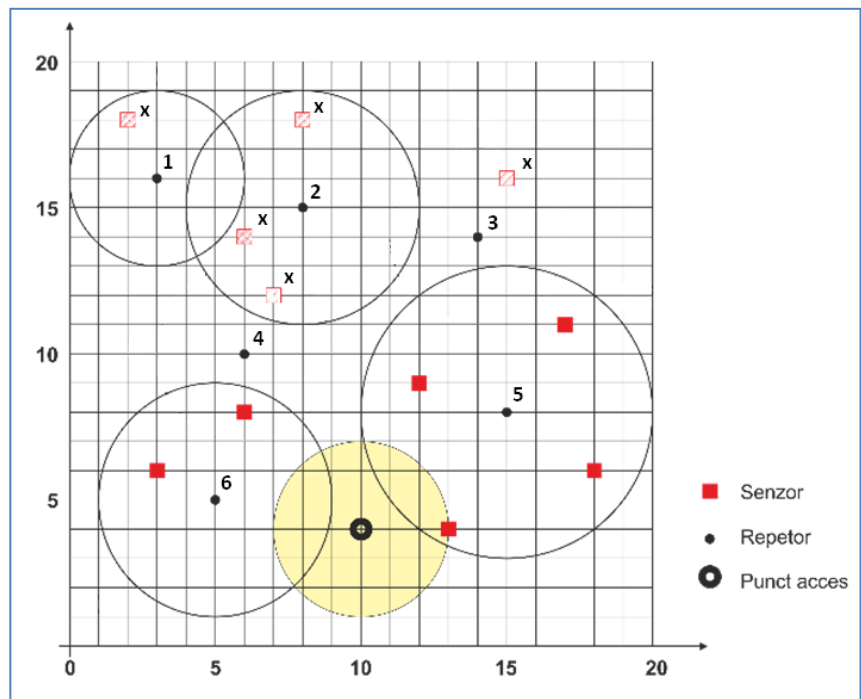


Figura 2

Explicație. Repetoarele 3 și 4 sunt nefuncționale (vezi Figura 2). Din această cauză, repetoarele 1 și 2 nu mai pot comunica cu punctul de acces. În consecință, senzorii marcați cu simbolul "x" au pierdut legătura cu punctul de acces. Evident, numărul acestor senzori este egal cu 5.