

Abori minimi de acoperire

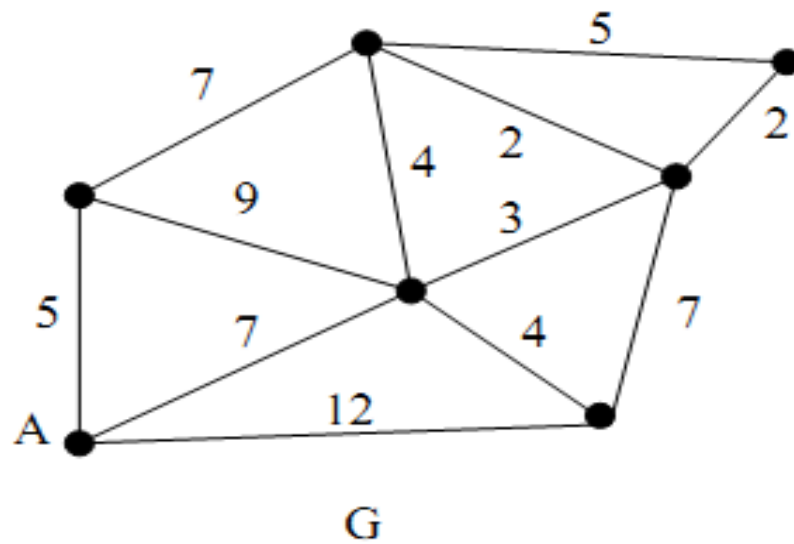
Ana Ozorchevici

ana.ozorchevici@gmail.com

Utilizare

- Organizare de rețele (telefonie, drumuri, electric)
- Aproximare pentru probleme NP-hard
- LDPC error correction
- STP
- Clustering

AMA



Algoritmul lui Prim

Initial: $V_{\text{new}} = \{x\}$, x un nod arbitrar (nod de start) din V , $E_{\text{new}} = \{\}$

Repetă până când $V_{\text{new}} = V$:

- Alege o muchie (u, v) cu costul minim astfel încât u este în V_{new} și v nu este
- Se adaugă v la V_{new} și (u, v) la E_{new}

Heap Fibonacci - operatii

- Aflarea cheii minime
- Inserare
- Extragerarea cheii minime
- Modificarea valorii cheii

Heap Fibonacci

- Radacinile arborilor sunt retinute printr-o lista circulara dublu inlantuita
- Copiii fiecarui nod sunt retinuti tot intr-o astfel de lista
- Se retine un pointer spre radacina arborelui care contine nodul cu cheia minima

Algoritmul lui Kruskal

- Se creeaza o padure de arbori F cu fiecare nod din graf reprezentand un arbore
- Se creeaza un set S continand toate muchiile grafului
- Cat timp S este nevid si F nu este un singur arbore
 - Se elimina muchia (u,v) de cost minim din S
 - Daca cele doua noduri ale muchiei fac parte din arbori diferiti cei doi arbori se unifica
 - Altfel se elimina muchia

Disjoint-set

- *Find*: Determinarea subsetului din care face parte un element
- *Union*: Unirea a doua subseturi intr-un singur subset.

Union

```
function Union(x, y)
  xRoot := Find(x)
  yRoot := Find(y)
  if xRoot == yRoot
    return
  // x and y are not already in same set. Merge them.
  if xRoot.rank < yRoot.rank
    xRoot.parent := yRoot
  else if xRoot.rank > yRoot.rank
    yRoot.parent := xRoot
  else
    yRoot.parent := xRoot
    xRoot.rank := xRoot.rank + 1
```

Find

```
function Find(x)
  if x.parent != x
    x.parent := Find(x.parent)
  return x.parent
```

Algoritmul lui Boruvka

Begin with an empty set of edges T

While the vertices of G connected by T are disjoint:

 Begin with an empty set of edges E

 For each component of T :

 Begin with an empty set of edges S

 For each vertex v in the component:

 Add the cheapest edge from v to a vertex in a different component of T to S

 Add the cheapest edge in S to E

 Add the resulting set of edges E to T .

AMA diferiti

Se pot obtine AMA diferiti folosind algoritmiul lui Kruskal si Prim?
Argumentati.

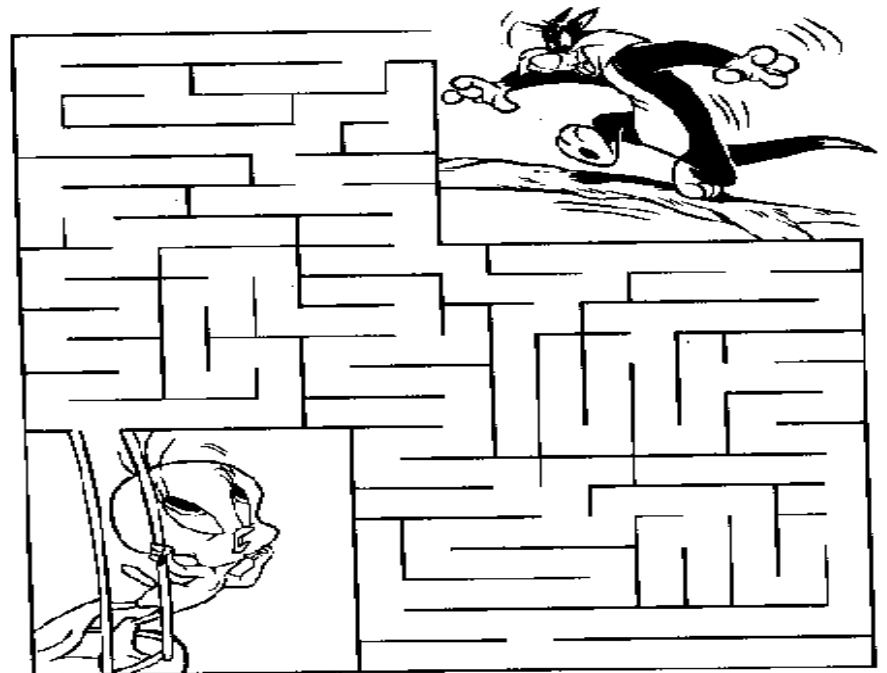
Cand AMA este unic?

AMA diferiti

- Un algoritm determina AMA (arborele **A**)
- Presupunem ca **A** nu este unic \Rightarrow Exista un alt AMA (**B**).
- Fie **e1** o muchie ce apartine lui **A** dar nu lui **B**.
- Cum **B** este AMA, $\{e1\} \cup B$ contine un ciclu **C** \Rightarrow **B** contine cel putin o muchie **e2** ce nu apartine lui **A** dar apartine lui **C**.
- Presupunem costul lui **e1** mai mic decat costul lui **e2**.
- Inlocuim **e2** cu **e1** in **B** \Rightarrow AMA $\{e1\} \cup B - \{e2\}$ care are un cost mai mic decat **B**.
- Contradiction (am presupus ca **B** este AMA).

Labirint

- Creati un labirint.
- Initial exista un grid format din celule inconjurate de 4 pereti.



Labirint - rezolvare

- O lista cu pereti si un set pentru celule fiecare celula intr-un subset
- Pentru fiecare perete (random):
 - Daca celulele despartite de perete fac parte din subseturi distincte:
 - Se inlatura peretele.
 - Se unesc cele doua subseturi.

Desen

Miruna deseneaza in plan N puncte. Ea defineste notiunea de arbore partial pentru un set de S puncte ca fiind o submultime de exact $S-1$ segmente avand capetele printre punctele date, astfel incat sa se poata ajunge dintr-un punct in oricare altul mergand pe segmentele alese.

Miruna defineste si notiunea de arbore partial de cost minim ca fiind acel arbore partial pentru care suma lungimilor segmentelor din submultimea aleasa sa fie minima.

Desen

Dupa ce deseneaza un punct nou in plan, Miruna doreste sa afle care este suma lungimilor segmentelor ce alcatuiesc arborele partial de cost minim pentru toate punctele desenate pana in acel moment.

<http://infoarena.ro/problema/desen>

Date de intrare	Date de iesire
-----------------	----------------

5	0.000000
0 0	2.000000
0 2	4.000000
2 0	6.000000
2 2	5.656854
1 1	

Desen - rezolvare

Pentru fiecare nou punct adaugat se aplica Kruskal sau Prim pe graful obtinut din AMA intial + muchiile rezultate din adaugarea noului nod

Online

Intr-o tara imaginara trebuie construita o retea de drumuri intre N orase.

Ministrul Transporturilor a apelat la M firme de constructii si le-a cerut sa estimeze costul unor drumuri care la randul lor sa lege doua orase, fara sa treaca prin altele.

Din cele M oferte care se refereau la M drumuri diferite , Ministrul a ales un numar minim de perechi de orase intre care se va construi un drum direct, astfel incat din orice oras sa se poata calatori in orice oras direct sau prin intermediul unor drumuri care trec prin alte orase cu cost minim.

Online

Construcția urmează să înceapă peste K săptămâni.

În această perioadă Ministrul Transporturilor a primit săptămânal câte o ofertă nouă, referitoare la alte posibile drumuri care vor lega în mod direct alte două orașe dar și la unele oferte deja, dar cu alte costuri.

Ministrul Transporturilor trebuie să actualizeze proiectul și să stabilească - săptămânal - care dintre drumuri să le construiască peste cele K săptămâni.

Determinați proiectul inițial. Apoi, pe baza ofertelor nou sosite pe parcursul celor K săptămâni, determinați câte un proiect nou astfel încât costul total al producției să fie minim.

<http://infoarena.ro/problema/online>

Online

Date de intrare:

5 7

1 2 1

2 3 7

1 5 5

3 4 3

4 5 4

1 3 2

2 4 6

2

3 5 2

1 4 3

Date de iesire

(costurile

proiectelor: a celui

initial, apoi ale celor

K actualizari)

10

8

8

Online - rezolvare

Initial se determina AMA

Pentru urmatorii k pasi:

- muchia curenta este intre doua noduri din AMA.
Comparam costul actual al muchiei cu cel din arbore:
cost mai mic => costul arborelui se imbunatateste cu
diferenta costurilor
altfel => cost neschimbat
- muchia nu face parte din arbore. Se sterge muchia de
cost maxim din arbore si se adauga muchia curenta

Karb

Se dă un graf neorientat simplu conex cu N noduri și M muchii. Muchiile au costul 0 sau 1 . Se cere să se determine un arbore de acoperire de cost exact K .

<http://infoarena.ro/problema/karb>

Date de intrare: Date de iesire:

6 8 3 1 3

1 3 1 3 4

1 2 0 5 6

2 3 1 5 2

3 4 1 4 2

2 4 0

2 5 0

4 6 1

5 6 1

Karb - rezolvare

1. cu muchiile de cost 0 descompune graful în componente conexe;
2. caută muchiile de cost 1 care unesc aceste componente, astfel încât să nu se formeze un ciclu între acestea;
3. consideră doar muchiile de cost 1 de la pasul anterior la care adaugă altele de același cost până ce vor fi în număr de K ;
4. adaugă muchii de cost 0 până ce se formează un arbore de acoperire;

Radiatie

Zaharel este cercatator de fizica nucleara si lucreaza la un complex de cercetare secret format din N laboratoare, numerotate convenabil cu numere de la 1 la N .

De asemenea, aceste laboratoare sunt conectate prin M tunele bidirectionale, de diverse lungimi.

Desi laboratoarele sunt concepute astfel incat cercetatorii sa nu fie expusi la radiatii, tunelele nu protejeaza complet, astfel incat daca un cercetator foloseste un anumit tunel de legatura va fi expus la un nivel de radiatii direct proportional cu lungimea tunelului.

Radiatie

Zaharel are de efectuat K drumuri intre diverse perechi de laboratoare. Stiind ca expunerea pentru un timp indelungat la radiatii are efecte daunatoare, el va alege mereu un drum in care lungimea maxima a unui tunel parcurs este minima.

Determinati care vor fi drumurile pe care le va alege Zaharel.

<http://infoarena.ro/problema/radiatie>

Radiatie

Date de intrare:

6 6 8
1 2 5
2 3 4
3 4 3
1 4 8
2 5 7
4 6 2
1 2
1 3
1 4
2 3
2 4
5 1
6 2
6 1

Date de iesire (lungimea
maxima minima)

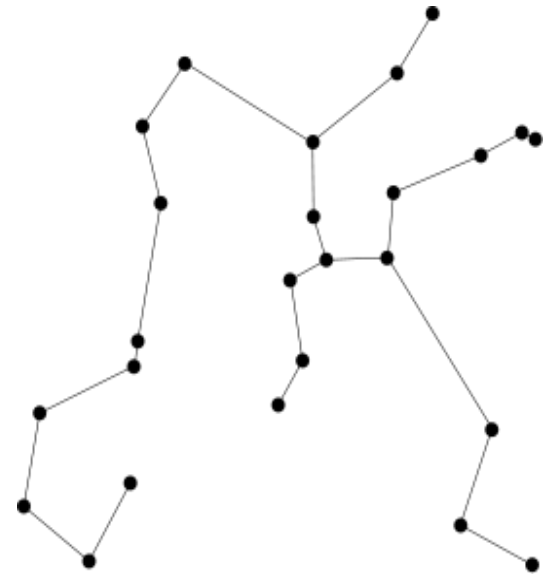
5
5
5
4
4
7
4
5

Radiatie-rezolvare

- Primul pas in rezolvarea problemei este construirea arborelui partial de cost minim
- Pentru a determina cea mai mare valoare a unei muchii pe drumul unic din arbore intre nodurile i si j folosim arborele care rezulta in urma algoritmului lui Kruskal, renuntand la euristica de compresie a drumului, si folosind doar euristica dupa rang
- Cum acest arbore are inaltimea $O(\log N)$, se poate folosi o parcurgere triviala pentru a raspunde la orice query.

Euclidean minimum spanning tree

Euclidean minimum spanning tree sau EMST este un AMA ce contine n puncte din plan unde costul muchiei dintre 2 puncte reprezinta distanta dintre acestea.



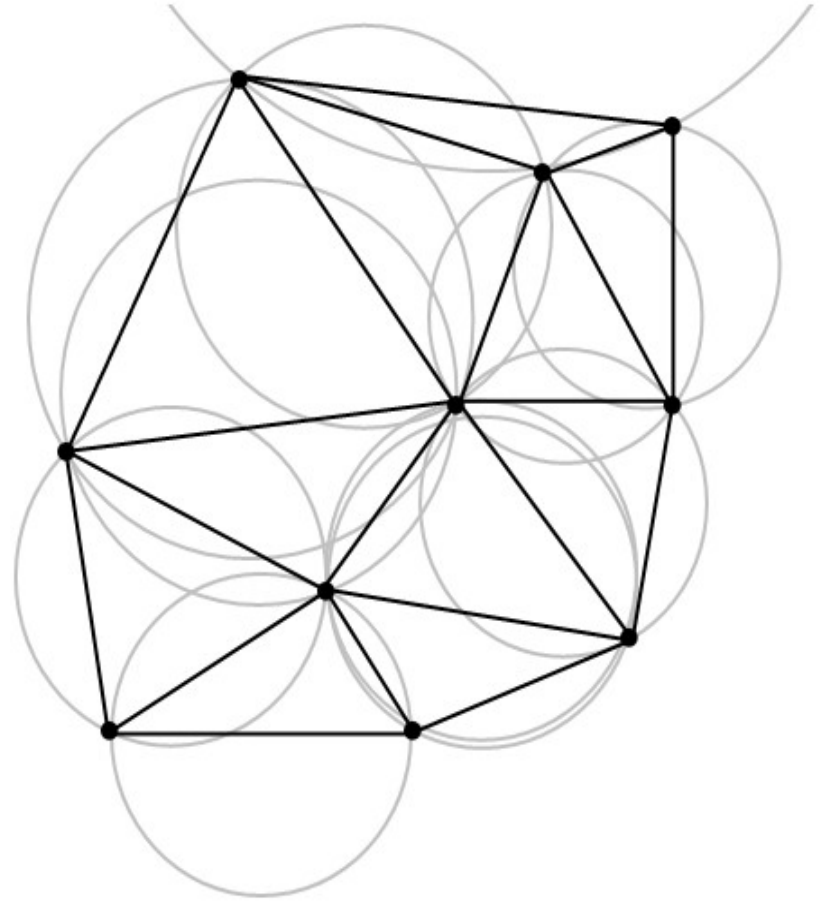
Euclidean minimum spanning tree

Se calculeaza triangularea Delaunay in $O(n \log n)$ time. Datorita faptului ca triangularea Delaunay determina un graf planar se vor genera doar $O(n)$ muchii.

Se aplica unul din algoritmi de determinare a AMA pe noul graf.

Triangularea Delaunay

Triangularea Delaunay pentru un set de puncte \mathbf{P} aflate in plan $DT(\mathbf{P})$ este o triangulare in care niciun punct din \mathbf{P} nu se afla in interiorul cercului circumscris oricarui triunghi din $DT(\mathbf{P})$.



Triangularea Delaunay

Metoda de a determina daca un punct D se afla in cercul circumscris triunghiului ABC este evaluarea determinatului:

$$\begin{vmatrix} A_x & A_y & A_x^2 + A_y^2 & 1 \\ B_x & B_y & B_x^2 + B_y^2 & 1 \\ C_x & C_y & C_x^2 + C_y^2 & 1 \\ D_x & D_y & D_x^2 + D_y^2 & 1 \end{vmatrix} = \begin{vmatrix} A_x - D_x & A_y - D_y & (A_x^2 - D_x^2) + (A_y^2 - D_y^2) \\ B_x - D_x & B_y - D_y & (B_x^2 - D_x^2) + (B_y^2 - D_y^2) \\ C_x - D_x & C_y - D_y & (C_x^2 - D_x^2) + (C_y^2 - D_y^2) \end{vmatrix} > 0$$

Resurse

- <http://infoarena.ro/>
- <http://infoarena.ro/problema/apm>
- <http://en.wikipedia.org/>