UPV/EHU—Informatika Ingeniaritza
DEA-II 2 PARTZIALA (2011-iv-14) - E taldea

Aukera: A B

Deiturak	1	2	3	4	5	6	7	8	

1. (1.5 pt) Demagun motxilaren 0/1 problema ondorengo ekuazio sistema erabiliz ebatzi dugula:

```
M(d,i)  = d edukiera duen motxilan 1..i elementuak sargarri izanik, lor
          litekeen irabazi maximoa
M(0,i)  = 0               ∀i
M(d,1)  = 0               if d<p(1)
        = b(1)            if p(1)≤d
M(d,i)  = max{b(i)+ M( d-p(i), i-1), M(d, i-1) }  if p(i)≤d && 1<i
        = M( d, i-1)}                             if p(i)>d && 1<i
```

eta objektuen markak dituen (1..N, 0..E) dimentsiodun matrizea kalkulatu dugula jarraian azaltzen diren balio zehatzekin:

E=18 p=(3,7,2,5,4,6) b=(3,8,4,4,3,8)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
3	0	0	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	1
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	1	1	1	1

Kalkula ezazu eskuz, zeintzuk diren zehazki motxilan sartu behar diren objektuak – horretarako erabili diren (1..N, 0..E) objektuen marken matrizeko gelaxken koordenatu zehatzak zeintzuk izan diren azalduaz- bai eta euren eramateak emango duen irabazi optimoa ere.

SOL:

Objektuak:
Koordenatuak:
Irabazia:

2. (2.25 pt) K euro ditugu eta b(1..n) balioak dituzten n objektu eskaintzen dizkigute. Ondorengo errekurtsio ekuazioak balio bat definitzen du:

```
Erosketak(d,i)  = d euro izanik eta 1..i objektuak erosgai, egin
       litezkeen erosketa desberdinen kopurua
Erosketak(0,i) = 1              ∀i
Erosketak(d,0) = 0              ∀d
Erosketak(d,i) = Erosketak(d,i-1)+Erosketak(d-b(i),i-1)  if b(i)≤d && 0<i
               = Erosketak(d,i-1)                        if b(i)>d && 0<i
```

Azaldutako funtzioa kodetzen duen algoritmo iteratibo bat idaztea eskatzen zaizu programazio dinamikoaren teknika erabiliz. Soluzioaren denbora ordena ere kalkula ezazu.

SOL: Kode iteratiboa eta ordena

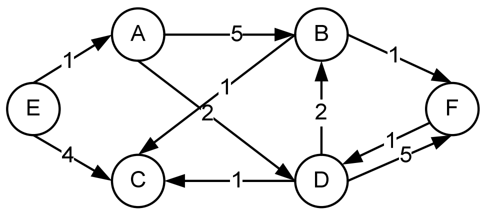
3. (2.25 pt) Lursail batean gertatutako sator izurriaren ostean, *B* pozoi-bola ezartzea erabaki dugu lursaileko hautazko *N* gunetan zehar banaturik. Pozoia zabaltzeak bere gaitzak dakartza, zeren bola desegin hala, lurrak pozoia xurgatzen du hura kutsatuz; are gehiago, puntu bakoitzaren xehetasunak eta utzitako bola kopurua kontuan izanik, kutsadura puntu bakoitzean desberdina da. Agronomo ingeniari batek *K(1..B,1..N)* taula osatu du, non *K(p,g)* osoko positiboak *p* pozoi-bola *g* gunean uzterakoan sortzen duen kutsadura adierazten duen.

   Agronomo ingeniariak orain *B* pozoi-bolak nola banatu behar dituen lursaileko *N* gunetan zehar erabaki behar du, lursailean produzitutako kutsaduren batura minimoa izan dadin.

   Soilik kutsadura minimoaren problema ebazten duten ekuazioak idaztea eskatzen zaizu, hemendik aurrerako pausoak programazio dinamikoaren teknika erabiliaz agronomoak ematen badakizki eta.

   SOL: Ekuazio sistema

4. (1.5 pt) Ondorengo grafoko A erpinatik gainontzeko erpinetarainoko distantzia minimoak kalkulatzea eskatzen zaizu Dijsktraren soluzio jalea aplikatuz. D distantzien bektorearen hasieraketa eta bira jale bakoitzean bai harrapatutako erpina bai eta D bektorearen eguneraketa ere islatzea eskatzen zaizu.



SOL:

	D taularen egoerak						
Grafoko erpina	A	B	C	D	E	F	Erpin hautatua
D taularen hasieraketa							
1 bira							
2 bira							
3 bira							
4 bira							

--- **A aukera:** Kruskal+Prim+Partiketa **--**

5. (1.5 pt) Ondorengo taulak Partiketaren egitura baten egoera zehaz bat jasotzen du:

1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*
-2	1	2	1	1	7	-1	7	-2	9	0	13	9

Ondoren azaltzen zaizkizun eragiten ondoren taula Partiketaren egoera zein den azal ezazu ematen zaizun egitura osatuz:

```
a) E1=Bilatu3(8); E2= Bilatu3(11); Bateratu3(E1, E2);
b) E1=Bilatu3(6); E2= Bilatu3(13); Bateratu3(E1, E2);
c) E1=Bilatu3(5); E2= Bilatu3(7); Bateratu3(E1, E2);
```

SOL:

1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*

--- A aukera: Kruskal+Prim+Partiketa **--**

6. (1.5 pt) Demagun *`PrimGrafo(G)`* deiak, G grafoaren hedapen zuhaitz minimo bat kalkulatu eta itzultzen duela (non euren pisuak positiboak diren). Bestalde, *`Dijkstra(G,e)`* deiak e erpinetik G grafoko beste erpinetaraino dauden distantzi minimoen taula kalkulatzen duela. Egia da *`Dijkstra(PrimGrafo(G),e)=`* *`Dijkstra(G,e)`*? Erantzuna arrazoitu.

   SOL:

--- B aukera --

7. (1.5 pt) Ekintza hautaketaren problema gogora ezazu (eskaerek esklusioan erabil lezaketen baliabide konpartituarena, non helburua zerbitzua lortzen duten eskaera kopurua maximizatzea den).

   a.- Ondorengo proposamenetatik aukeratu prozedura hautesle zuzena:
   (1) gutxien irauten duen ekintza
   (2) denbora ardatzean gutxien teilakatzen dena
   (3) beranduen hasten dena
   (4) lehenen hasten den

   b.- hautatu duzun prozedura hauteslea integratua duen algoritmo jaleak zer itzuliko du ondorengo eskaerak ematen dizkiogunean? {[8:30, 9:30), [12:00, 12:45), [11:30, 12:15), [11:00, 12:30), [10:00, 10:50) }

   SOL:

--- B aukera --

8. (1.5 pt) Dijkstraren algoritmoaren egokitu ezazu erpin batetik beste erpinetaraino doazen bide motzenen kostua kalkulatzeaz gain, erpin bakoitzeraino dagoen bide motzenen kopurua kalkula dezan.

```
procedure  DIJKSTRA ( G: in GRAFOA;  D: out TAULA) is

...P: Integer:=G'LENGTH(1);

begin

   HASIERATU2_P(G,H);

   for K in  G'FIRST(1)+1..G.'LAST(1) loop  -- Oro har: 2..P

         D(K):= G(1,K);

   end loop;

   for  I in  P-2  loop

      GertuenDagoenErpina(H,D,X);

      HautagaietatikKendu(H,X);

      HK:= H;

      for J in 1..ErpinKopurua(H) loop

         Y:= LehenengoaL(HK);

         AurreraEgin(HK);

         if D(Y) > D (X)+G(X,Y)

         then D (Y) := D(X)+G(X,Y);

         end if;

      end loop;

   end loop;

end DIJKSTRA ;
```

UPV/EHU—Informatika Ingenieritza
DEA-II 2 PARTZIALA (2011-iv-14) - E taldea
SOLUZIO BAT

1. (1.5 pt) Demagun motxilaren 0/1 problema ondorengo ekuazio sistema erabiliz ebatzi dugula:

```
M(d,i)  = d edukiera duen motxilan 1..i elementuak sargarri izanik, lor
          litekeen irabazi maximoa
M(0,i)  = 0                ∀i
M(d,1)  = 0                if d<p(1)
        = b(1)             if p(1)≤d
M(d,i)  = max{b(i)+ M( d-p(i), i-1), M(d, i-1) }   if p(i)≤d && 1<i
        = M( d, i-1)}                               if p(i)>d && 1<i
```

eta objektuen markak dituen (1..N, 0..E) dimentsiodun matrizea kalkulatu dugula jarraian azaltzen diren balio zehatzekin:

E=18 p=(3,7,2,5,4,6) b=(3,8,4,4,3,8)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
3	0	0	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	1
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
6	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	1	1	1	1

Kalkula ezazu eskuz, zeintzuk diren zehazki motxilan sartu behar diren objektuak – horretarako erabili diren (1..N, 0..E) objektuen marken matrizeko gelaxken koordenatu zehatzak zeintzuk izan diren azalduaz- bai eta euren eramateak emango duen irabazi optimoa.

SOL:

Objektuak: 1,2,3,6
Koordenatuak: illunduak
Irabazia: 23

2. (2.25 pt) K euro ditugu eta b(1..n) balioak dituzten n objektu eskaintzen dizkigute. Ondorengo errekurtsio ekuazioak balio bat definitzen du:

```
Erosketak(d,i)  =  d  euro  izanik  eta  1..i  objektuak  erosgai,  egin
       litezkeen erosketa desberdinen kopurua
Erosketak(0,i) = 1              ∀i
Erosketak(d,0) = 0              ∀d
Erosketak(d,i) = Erosketak(d,i-1)+Erosketak(d-b(i),i-1)  if b(i)≤d && 0<i
               = Erosketak(d,i-1)                        if b(i)>d && 0<i
```

Azaldutako funtzioa kodetzen duen algoritmo iteratibo bat idaztea eskatzen zaizu programazio dinamikoaren teknika erabiliz. Soluzioaren denbora ordena ere kalkula ezazu.

SOL: Kode iteratiboa eta ordena

```
procedure Erosketak (K: in Integer; B; in array (1..n) of integer;
                     Konbinazioak: out Integer) is

   MErosi: array of(0..K,0..n) of integer;
begin
   for I in 0..N loop MErosi(0,I):= 1; end loop;
   for D in 1..K loop MErosi(D, 0):= 0; end loop;
   for I in 1..N loop
         for D in 1..K loop
               if (b(I)≤D) then
                     MErosi(D,I):= MErosi(D- b(I),I-1)+ MErosi(D,I-1);
               else  MErosi(D,I):= MErosi(D,I-1);
               end if;
         end loop;
   end loop;
Konbinazioak:= Merosi(K,N);

Begi bistakoa da soluzioa ordena O(N*K) dela.
```

3. (2.25 pt) Lursail batean gertatutako sator izurriaren ostean, *B* pozoi-bola ezartzea erabaki dugu lursaileko hautazko *N* gunetan zehar banaturik. Pozoia zabaltzeak bere gaitzak dakartza, zeren bola desegin hala, lurrak pozoia xurgatzen du hura kutsatuz; are gehiago, puntu bakoitzaren xehetasunak eta utzitako bola kopurua kontuan izanik, kutsadura puntu bakoitzean desberdina da. Agronomo ingeniari batek *K(1..B,1..N)* taula osatu du, non *K(p,g)* osoko positiboak *p* pozoi-bola *g* gunean uzterakoan sortzen duen kutsadura adierazten duen.

Agronomo ingeniariak orain *B* pozoi-bolak nola banatu behar dituen lursaileko *N* gunetan zehar erabaki behar du, lursailean produzitutako kutsaduren batura minimoa izan dadin.

Soilik kutsadura minimoaren problema ebazten duten ekuazioak idaztea eskatzen zaizu, hemendik aurrerako pausoak programazio dinamikoaren teknika erabiliaz agronomoak ematen badakizki eta.

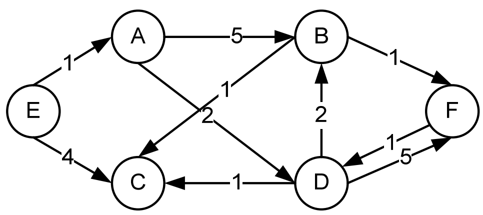
SOL: Ekuazio sistema

Kutsadura da optimizatu behar dena, hortaz, programazio dinamikoaren ekuazio sistemak kutsadura minimoa kalkulatu beharko du. Ekuazio sistema honen parean, zeintzuk bolak zein zulotan sartu behar diren, objektuen berreskurapenaren atala litzateke. Hona hemen ekuazio sistema:

KutsaduraTxiki(b,g): b pozoi-bola 1..g gunetan banatzerakoan sortarazi litekeen kutsadura minimoa.
KutsaduraTxiki(0,g)=0
KutsaduraTxiki(b,1)=K(b,1)   b>0
KutsaduraTxiki(b,g)=min{   KutsaduraTxiki(b,g-1)
K(1,g)+KutsaduraTxiki(b-1,g-1)
K(2,g)+KutsaduraTxiki(b-2,g-1)
…
K(b,g)+0 }

Oharra ez du balio jarraiko kasu orokorrak, bolen kutsadura uniformea ez baita:
KutsaduraTxiki(b,g)=min{   KutsaduraTxiki(b,g-1), K(1,g)+KutsaduraTxiki(b-1,g)};
Hots, printzipioz K(1,g)+K(1,g) =/= K(2,g) gerta litekeelako.

4. (1.5 pt) Ondorengo grafoko A erpinatik gainontzeko erpinetarainoko distantzia minimoak kalkulatzea eskatzen zaizu Dijsktraren soluzio jalea aplikatuz. D distantzien bektorearen hasieraketa eta bira jale bakoitzean bai harrapatutako erpina bai eta D bektorearen eguneraketa ere islatzea eskatzen zaizu.



SOL:

	D taularen egoerak						
Grafoko erpina	A	B	C	D	E	F	Erpin hautatua
D taularen hasieraketa		5	∞	2	∞	∞	
1 bira		4	3	2	∞	7	D
2 bira		4	3	2	∞	7	C
3 bira		4	3	2	∞	5	B
4 bira		4	3	2	∞	5	F

--- **A aukera:** Kruskal+Prim+Partiketa **--**

5. (1.5 pt) Ondorengo taulak Partiketaren egitura baten egoera zehaz bat jasotzen du:

1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*
-2	1	2	1	1	7	-1	7	-2	9	0	13	9

Ondoren azaltzen zaizkizun eragiten ondoren taula Partiketaren egoera zein den azal ezazu ematen zaizun egitura osatuz:

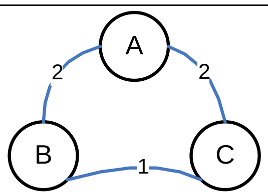
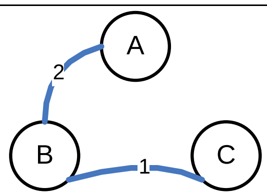
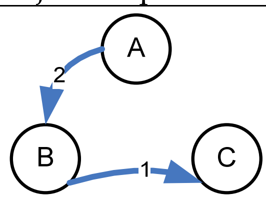
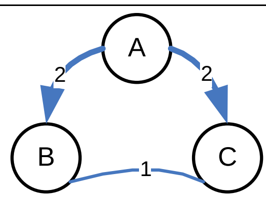
```
d) E1=Bilatu3(8); E2= Bilatu3(11); Bateratu3(E1, E2);
e) E1=Bilatu3(6); E2= Bilatu3(13); Bateratu3(E1, E2);
f) E1=Bilatu3(5); E2= Bilatu3(7); Bateratu3(E1, E2);
```

SOL:

1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*
-3	1	2	1	1	7	**9**	7	**1**	9	**7**	13	9

6. (1.5 pt) Demagun `PrimGrafo(G)` deiak, G grafoaren hedapen zuhaitz minimo bat kalkulatu eta itzultzen duela (non euren pisuak positiboak diren). Bestalde, `Dijkstra(G,e)` deiak e erpinetik G grafoko beste erpinetaraino dauden distantzi minimoen taula kalkulatzen duela. Egia da `Dijkstra(PrimGrafo(G),e)=` `Dijkstra(G,e)`? Erantzuna arrazoitu.

SOL:

grafoa	Primen HZM, A-tik hasiz	Aurrekoari Dijkstra aplikatuz	Dijkstra(G,A)
			

--- **B aukera:** Kruskal+Prim+Partiketa **GABE--**

7. (1.5 pt) Ekintza hautaketaren problema gogora ezazu (eskaerek esklusioan erabil lezaketen baliabide konpartituarena, non helburua zerbitzua lortzen duten eskaera kopurua maximizatzea den).

   a.- Ondorengo proposamenetatik aukeratu prozedura hautesle zuzena:
   (1) gutxien irauten duen ekintza
   (2) denbora ardatzean gutxien teilakatzen dena
   (3) beranduen hasten dena
   (4) lehenen hasten den

   b.- hautatu duzun prozedura hauteslea integratua duen algoritmo jaleak zer itzuliko du ondorengo eskaerak ematen dizkiogunean? {[8:30, 9:30), [12:00, 12:45), [11:30, 12:15), [11:00, 12:30), [10:00, 10:50) }

   SOL: (3) Beranduen hasten dena

   {[8:30, 9:30), [12:00, 12:45), ~~[11:30, 12:15)~~, ~~[11:00, 12:30)~~, [10:00, 10:50) }

8. (1.5 pt) Dijkstraren algoritmoaren egokitu ezazu erpin batetik beste erpinetaraino doazen bide motzenen kostua kalkulatzeaz gain, erpin bakoitzeraino dagoen bide motzenen kopurua kalkula dezan.

   SOL: BKop(X)= Xraino dauden bide motz kopurua jasotzen du

```
procedure  DIJKSTRA ( G: in GRAFOA;  D: out TAULA) is
...P: Integer:=G'LENGTH(1);
begin
   HASIERATU2_P(G,H);
   for K in  G'FIRST(1)+1..G.'LAST(1) loop  -- Oro har: 2..P
         D(K):= G(1,K);
         If D(K)≠∞ then BKop(K)=1; else BKop(K)=0; end if;
   end loop;
   for  I in  P-2  loop
      GertuenDagoenErpina(H,D,X);
      HautagaietatikKendu(H,X);
      HK:= H;
      for J in 1..ErpinKopurua(H) loop
         Y:= LehenengoaL(HK);
         AurreraEgin(HK);
         if D(Y) > D (X)+G(X,Y)
         then D (Y) := D(X)+G(X,Y); BKop(Y):=BKop(X);
         elsif D(Y) = D (X)+G(X,Y) then BKop(Y):=BKop(Y)+ BKop(X);
         end if;
      end loop;
   end loop;
end DIJKSTRA ;
```