**Operációkutatás II, 7. gyakorlat** 2011. március 30.

1. Tekintsük páratlan $n$-re a következő feladatot:

$$\begin{aligned} \max \quad & -x_{n+1} \\ 2x_1 + \cdots + 2x_n + x_{n+1} &= n \\ x &\in \{0,1\}^{n+1} \end{aligned}$$

   Mutassuk meg, hogy az órán látott korlátozás és szétválasztás algoritmus ennél a feladatnál mindenképpen exponenciálisan sok részfeladatot néz meg.

2. Adott egy fa, és minden csúcsának egy nemnegatív súlya. Adjunk algoritmust maximális súlyú stabil csúcshalmaz megtalálására. (Egy csúcshalmaz stabil, ha nem tartalmaz szomszédos csúcsokat.)

3. Egy szövegszerkesztő program egy dokumentumot oldalakra akar osztani. A dokumentum $n$ darab egymás utáni elemből (szavakból, ábrákból) áll. Minden $1 \le i \le j \le n$-re adott hogy az $i$-edik elemmel kezdődő és $j$-edik elemmel végződő oldal mennyire szép: $c_{ij}$. A dokumentum szépségét úgy kapjuk, hogy összeadjuk az oldalak szépségét. Adjunk dinamikus programozási algoritmust, ami meghatározza a legszebb beosztást.

4. Adott két DNS szekvencia, $S_1$ és $S_2$, amik nem feltétlenül egyforma hosszúak. Mindkét szekvencia az $\{A, C, G, T\}$ halmazból vett betűk sorozata. Arra vagyunk kíváncsiak, hogy mennyire nehéz az $S_1$ szekvenciát átalakítani az $S_2$ szekvenciává. A megengedett műveletek a következők:

   - A szekvenciába valahova beszúrunk egy elemet. Ennek nehézsége $\alpha$.
   - A szekvenciából valamelyik elemet töröljük. Ennek nehézsége $\beta$.
   - A szekvencia $j$-edik elemét átalakítjuk $x$-ről $y$-ra, ahol $x, y \in \{A, C, G, T\}$. Ennek nehézsége $c_{xy}$, tehát a betűktől függ.

   Az átalakítás nehézsége a műveletek nehézségének összege. Adjunk dinamikus programozási algoritmust, ami kiszámolja a legkönnyebb átalakítást.

**Operációkutatás II, 7. gyakorlat** 2011. március 30.

1. Tekintsük páratlan $n$-re a következő feladatot:

$$\max \quad -x_{n+1}$$
$$2x_1 + \cdots + 2x_n + x_{n+1} = n$$
$$x \in \{0,1\}^{n+1}$$

   Mutassuk meg, hogy az órán látott korlátozás és szétválasztás algoritmus ennél a feladatnál mindenképpen exponenciálisan sok részfeladatot néz meg.

2. Adott egy fa, és minden csúcsának egy nemnegatív súlya. Adjunk algoritmust maximális súlyú stabil csúcshalmaz megtalálására. (Egy csúcshalmaz stabil, ha nem tartalmaz szomszédos csúcsokat.)

3. Egy szövegszerkesztő program egy dokumentumot oldalakra akar osztani. A dokumentum $n$ darab egymás utáni elemből (szavakból, ábrákból) áll. Minden $1 \leq i \leq j \leq n$-re adott hogy az $i$-edik elemmel kezdődő és $j$-edik elemmel végződő oldal mennyire szép: $c_{ij}$. A dokumentum szépségét úgy kapjuk, hogy összeadjuk az oldalak szépségét. Adjunk dinamikus programozási algoritmust, ami meghatározza a legszebb beosztást.

4. Adott két DNS szekvencia, $S_1$ és $S_2$, amik nem feltétlenül egyforma hosszúak. Mindkét szekvencia az $\{A, C, G, T\}$ halmazból vett betűk sorozata. Arra vagyunk kíváncsiak, hogy mennyire nehéz az $S_1$ szekvenciát átalakítani az $S_2$ szekvenciává. A megengedett műveletek a következők:

   - A szekvenciába valahova beszúrunk egy elemet. Ennek nehézsége $\alpha$.
   - A szekvenciából valamelyik elemet töröljük. Ennek nehézsége $\beta$.
   - A szekvencia $j$-edik elemét átalakítjuk $x$-ről $y$-ra, ahol $x, y \in \{A, C, G, T\}$. Ennek nehézsége $c_{xy}$, tehát a betűktől függ.

   Az átalakítás nehézsége a műveletek nehézségének összege. Adjunk dinamikus programozási algoritmust, ami kiszámolja a legkönnyebb átalakítást.