Problém výberu aktivít

Dané: n aktivít $A_1, \ldots, A_n$

aktivita A_i: začiatok s_i, koniec f_i

Úloha: Vyber aktivity, ktoré sa **neprekrývajú**

Najväčší možný počet!

Možné riešenie?

1. Vyber aktivitu **s najmenším počtom konfliktov**
2. Vymaž všetky aktivity, ktoré sa s ňou prekrývajú
3. Opakuj

```
<--->   <----->   <----->   <--->
<--->        <------->      <--->
   <------->           <------->
   <------->           <------->
```

Greedy algoritmus pre problém výberu aktivít

Vždy vyber aktivitu, ktorá **končí najskôr**

```
Sort all activities by their finishing time
(now f[1]<=f[2]<=...<=f[n])

last_activity_end:=-infinity;

for i:=1 to n
  if (s[i]>=last_activity_end) then
     output activity (s[i],f[i]);
     last_activity_end:=f[i];
```

Časová zložitosť: $\Theta(n \log n)$

```
<--->   <----->   <----->   <--->
<--->        <------->      <--->
   <------->           <------->
   <------->           <------->
```

Dôkaz správnosti

Tvrdenie: Nech náš algoritmus vyberie aktivity $G = (G_1, G_2, \ldots, G_k)$. Potom pre ľubovoľné $0 \le \ell \le k$ existuje **optimálne riešenie** tvaru $O = (G_1, \ldots, G_\ell, O_{\ell+1}, \ldots, O_m)$.

Dôkaz indukciou vzhľadom na premennú ℓ:

Báza indukcie: Ak $\ell = 0$, platí triviálne.

Dôkaz správnosti

Tvrdenie: Nech náš algoritmus vyberie aktivity $G = (G_1, G_2, \dots, G_k)$. Potom pre ľubovoľné $0 \le \ell \le k$ existuje **optimálne riešenie** tvaru $O = (G_1, \dots, G_\ell, O_{\ell+1}, \dots, O_m)$.

Dôkaz indukciou vzhľadom na premennú ℓ:

Indukčný krok: Nech platí pre ℓ
$\Rightarrow$ optimálne riešenie $O = (G_1, \dots, G_\ell, O_{\ell+1}, O_{\ell+2}, \dots, O_m)$

$f_{O_{\ell+1}} \le s_{O_{\ell+2}}$ (správne zoradenie aktivít)
$f_{G_{\ell+1}} \le f_{O_{\ell+1}}$ (lebo vyberáme najmenší koniec)
$\Rightarrow f_{G_{\ell+1}} \le s_{O_{\ell+2}}$
$\Rightarrow$ môžeme vymeniť $O_{\ell+1}$ za $G_{\ell+1}$!

Teda: $O' = (G_1, \dots, G_{\ell+1}, O_{\ell+2}, \dots, O_m)$ je optimálne riešenie, ktoré súhlasí s prvými $\ell + 1$ výbermi!

Typický greedy algoritmus

- Každé riešenie získame pomocou postupnosti rozhodnutí.
- Nie všetky rozhodnutie vedú k optimálnemu riešeniu.
- V každom kroku:
  - Ováhuj všetky možné rozhodnutia pomocou nejakej váhovacej funkcie.
  - Vyber rozhodnutie s najväčšou váhou.

“Vzor” dôkazu správnosti greedy algoritmu

Lema: Predpokladajme, že greedy algoritmus vráti riešenie G. Potom existuje optimálne riešenie, ktoré sa s riešením G zhoduje na prvých k voľbách.

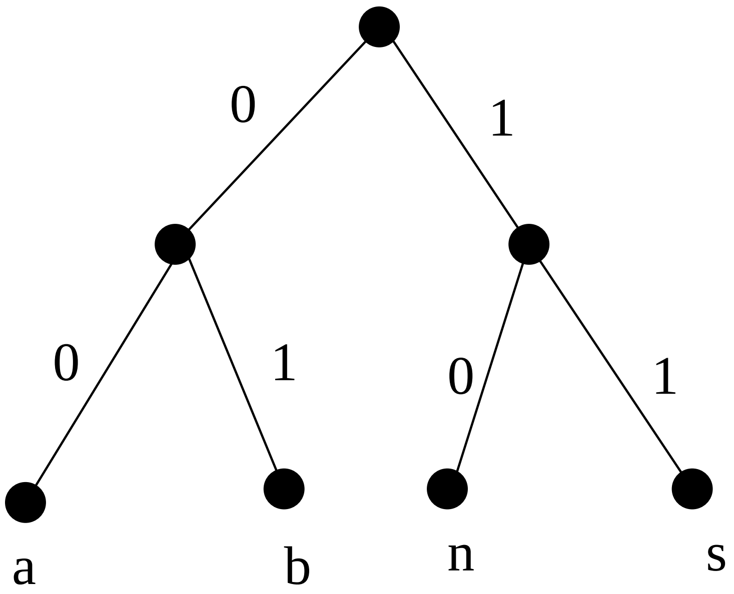
Dôkaz: Matematickou indukciou podľa k.

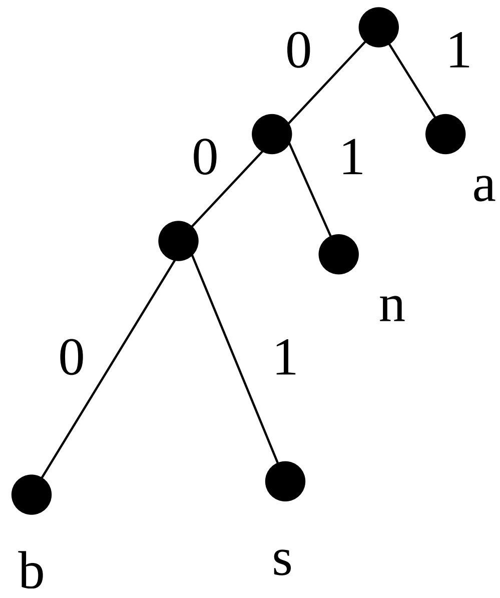
Báza indukcie. Pre $k = 0$ – ľubovoľné optimálne riešenie.

Indukčný krok. (Prepokladajme, že sme neurobili chybu pri prvých k voľbách, potom aj $(k + 1)$-vá voľba je OK.)

- Predpokladajme, že existuje optimálne riešenie OPT, ktoré sa zhoduje s G na prvých k voľbách.
- Vyrobíme riešenie OPT':
  - OPT' má rovnakú hodnotu ako OPT (a preto je tiež optimálne)
  - OPT' súhlasí s G na jednej ďalšej $(k + 1)$-vej voľbe.

Huffmanove prefixové kódy





Pre daný reťazec, rôzne stromy dávajú rôznu dĺžku kódovania.

Vieme nájsť taký strom (prefixový kód), ktorý dokáže najviac skomprimovať daný reťazec S?

Greedy algoritmus pre Huffmanov strom

```
Compute frequencies of all characters in S

F:=empty-forest;
for all characters x in the alphabet do
  T:=new leaf(x);
  add T to F;

while F contains more than one tree do
  T1:=extract tree with minimum frequency from F;
  T2:=extract tree with minimum frequency from F;
  T:=new tree where T1 is a left child
               and  T2 is a right child;
  add T to F;

return F;
```

Dôkaz správnosti greedy algoritmu pre Huffmanove stromy

Tvrdenie: Nech $F = \{T_1, T_2, \ldots, T_k\}$ je les, ktorý greedy algoritmus dostane po i krokoch.

Potom existuje **optimálny kódovací strom**, ktorý obsahuje stromy $T_1, T_2, \ldots, T_k$ ako podstromy.

Dôkaz indukciou podľa i:

Báza indukcie: Po 0 krokoch (inicializácia greedy algoritmu) je F jednoducho množina jednotlivých vrcholov zodpovedajúcich písmenám abecedy
$\Rightarrow$ platí triviálne

Indukčný krok: Nech po i krokoch máme $F = \{T_1, T_2, \ldots, T_k\}$ (bez ujmy na všeobecnosti, usporiadané od najmenšej frekvencie) Z indukčného predpokladu: existuje strom OPT, ktorý obsahuje stromy z F ako podstromy.

Čo urobí Greedy algoritmus? Spojí stromy T_1 a $T_2 \Rightarrow T$

Prípad 1: OPT obsahuje T ako podstrom

Prípad 2: OPT neobsahuje T ako podstrom

Keďže $f(T_1) \leq f(T_2) \Rightarrow d_1 \geq d_2$

čo by sa stalo ak by sme vymenili A a T_2?

– ak $d_1 = d_2$, potom ok; predpokladajme $d_1 > d_2$

– $f(A) < f(T_2)$ nemôže byť

– ak $f(A) > f(T_2)$: dostali by sme lepší strom $\Rightarrow$ spor!

– ak $f(A) = f(T_2)$: rovnako dobrý strom, ale obsahuje T!

Huffmanove stromy—záver

- Navrhli sme greedy algoritmus, ktorý nájde pre text s danými frekvenciami písmen kódovací strom
- Indukciou sme ukázali, že optimálny kódovací strom obsahuje "medzivýsledky" greedy algoritmu ako podstromy
- To platí aj po skončení greedy algoritmu, keď už máme len jeden výsledný strom $\Rightarrow$ tento strom je optimálny

Časová zložitosť?

Závisí od implementácie "lesa" (potrebujeme operácie: pridať nový strom nejakej veľkosť, vybrať najmenší strom):

- jednoduchý zoznam: $O(m + n^2)$
- prioritná fronta: $O(m + n \log n)$

Zhrnutie prednášky

- Greedy algoritmy sú veľmi ľahké na návrh a implementáciu
- Často ťažké dokázať, že algoritmus je správny (dá vždy optimálne riešenie)
- Dôkaz obvykle indukciou:
  - Začneme s nejakým optimálnym riešením
  - Postupne ho v rámci krokov indukcie "pretvárame" tak, aby súhlasilo s greedy riešením viac a viac
  - V žiadnom z krokov riešenie nezhoršíme, takže aj výsledné riešenie (identické s greedy riešením) je optimálne