**PROBLEM STABILNIH SOSEDOV**

Problem stabilnih sosedov se uporablja v matematiki, ekonomiji in računalništvu. Predvsem na področju kombinatorike, teorije iger in algoritmov.

V osnovi je podoben problemu stabilnih porok, vendar pa ga z njim ne smemo zamenjati. Saj pri problemu stabilnih sosedov spol parov ni pomemben in iščemo pare med vsemi kandidati.

Problem stabilnih sosedov, je problem kako najti stabilno ujemanje za sodo veliko množico. Ujemanje je ločitev niza na disjunktne pare (sosede). Ujemanje je stabilno, če ima vsak element svojega soseda, in je to za oba najboljša izbira. Obstajajo tudi problemi stabilnih sosedov, pri katerih ni mogoče najti rešitve. Ter primeri, ko rešitev ne zadovolji željam vseh sosedov.

**Irvingov algoritem**

Prvi učinkovit algoritem je napisal Robert Irving leta 1985. Časovna zahtevnost algoritma je $O(n^{2})$.

Algoritem je sestavljen iz dveh faz. V prvi fazi, udeleženci razvrstijo vse možne kandidate, glede na njihove želje, v padajočem vrstnem redu. Nato eden za drugim vprašajo kandidata, ki je prvi na njihovem seznamu, če bi bil njegov sostanovalec. Če vprašana osebe še nima sostanovalca prošnjo sprejme. V primeru, da pa je že predhodno sprejela povabilo, sedaj primerja obe možnosti, ter se odloči za tisto, ki je višje na njenem seznamu želj. Drugo osebo pa zavrne. V tem primeru morata oba kandidata prečrtati imena drug drugega na svojem seznamu. Ta postopek ponavljamo, dokler nimajo vse osebe svojega para. V primeru, da kateri od kandidatov ostane brez možnosti za par, to pomeni, da so vsi kandidati na njegovem seznamu prečrtani, torej problem nima stabilnih ujemanj.

V drugem delu prve faze se odstrani iz kandidatovih seznamov vse možne kandidate, ki so slabši oziroma nižje na seznamu od trenutno izbranega kandidata. To storimo tako, da gremo po vrsti pogledat seznam vsakega od kandidatov, ter simetrično odstranjujemo kandidate. To pomeni, da če kandidat $a\_{i}$ izloči kandidata $a\_{j}$ s svojega seznama mora tudi $a\_{j}$ izločiti $a\_{i} $s svojega seznama.
Če ima kateri od kandidatov na koncu prve faze na svojem seznamu samo še eno ne prečrtano ime, to pomeni, da smo mu že našli par. Taka oziroma take kandidate izločimo iz nadaljnjega postopka.

Druga faza je sestavljena iz rotacijskega izločanja. Rotacija je določena z dvema seznamoma: p in q. Da poiščemo tako rotacijo potrebujemo začetno osebo $p\_{0}$, ki ima vsaj dva kandidata na svojem seznamu želja. Sedaj rekurzivno definiramo $q\_{i}$, ki je drugi na $p\_{i}$-ovem seznamu in $p\_{i+1},$ ki je zadnji na seznamu od $q\_{i}$, dokler se ne ponovi nek $p\_{j}$. Rešitev je seznam parov $(p\_{i+1}, q\_{i})$. Ko imamo pare lahko začnemo z izločanjem. To poteka tako, da $q\_{i}$ izloči $p\_{i+1}$ in obratno, za vsak $i$. Če seznam enega od kandidatov, na enem izmed korakov, postane prazen, problem nima stabilnih ujemanj. V nasprotnem primeru pa imamo seznam stabilnih ujemanj. Če ima kateri od kandidatov, na svojem seznamu več kot eno osebo, je potrebno postopek rotacije ponoviti.

Postopek iskanja stabilnih ujemanj je zaključen, ko ima vsak od kandidatov na svojem seznamu natanko eno možno izbiro.