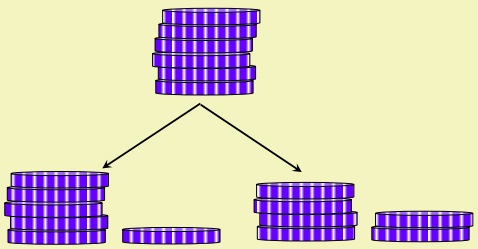
V. Kétszemélyes játékok

Teljes információjú, véges, zéró összegű kétszemélyes játékok

- Két játékos lép felváltva adott szabályok szerint.
- Mindkét játékos ismeri a maga és az ellenfele összes választási lehetőségét, és azok következményeit.
- Mindkét játékos minden lépésében véges számú lehetőség közül választhat;
  minden játszma véges lépésben véget ér.
- Amennyit az egyik játékos nyer, annyit veszít a másik. (Legegyszerűbb változatban: egyik nyer, másik veszít, esetleg lehet döntetlen is)

Grundy mama játéka

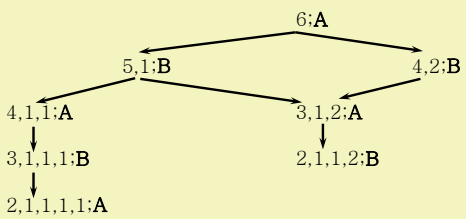


Állapottér-reprezentáció

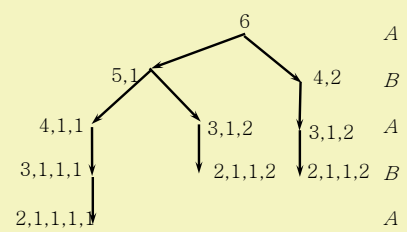
- állapot - állás + soron következő játékos
- művelet - lépés
- kezdő állapot - kezdőállás + kezdő játékos
- célállapot - végállás
  (nyerő, vesztő vagy döntetlen)

Egy játszma egy kezdőállapotból célállapotba vezető (mindig véges) műveletsorozat

Grundy mama játékgráfja



Grundy mama játékfája



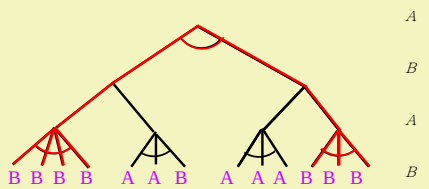
Játékfa

- csúcs - állás (egy állásnak több csúcs is megfelelhet)
- szint - játékos
- él - lépés (szintről szintre)
- gyökér - kezdőállás(kezdő játékos)
- levél - végállások
- ág - játszma

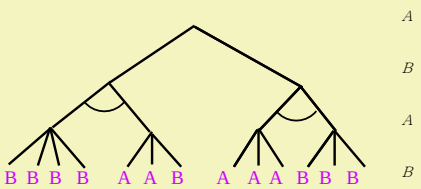
Nyerő stratégia

- A győztes (vagy nem-vesztő) stratégia egy olyan elv, amelyet betartva az ellenfél minden lépésére tudunk olyan választ adni, hogy megnyerjük (ne veszítsük el) a játékot.
- A győztes stratégia NEM egyetlen győztes játszma, hanem olyan győztes játszmák összessége, amelyek közül az egyiket biztos végig tudjuk játszani.

*Nyerő stratégia keresése az **B** játékos ÉS/VAGY fájában*



*Nyerő stratégia keresése az **A** játékos ÉS/VAGY fájában*



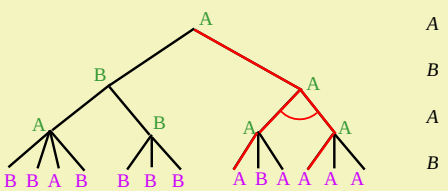
- Csak az egyik játékosnak lehet nyerő stratégiája.

Megjegyzés

- A nyerő (nemvesztő) stratégia egy ilyen megfelelő ÉS/VAGY fabeli megoldásgráffal ábrázolható, ahol a *T*-beli célcsúcsok a nyerő (nemvesztő) végállások.
- A nyerő stratégia keresése egy útkeresési probléma.

Tétel

- Egyik játékos számára mindig létezik nyerő stratégia (nem vesztő stratégia).



Részleges játékfa-kiértékelés

- Minimax algoritmus
- Negamax algoritmus
- Átlagoló kiértékelés
- Váltakozó mélységű
- Szelektív kiértékelés
- Alfabéta algoritmus

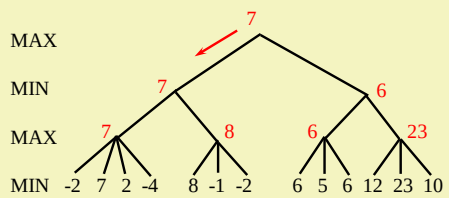
Kiértékelő függvény

- Minden esetben szükségünk van egy olyan heurisztikára, amely megbecsüli, hogy egy állás mennyire ígéretes.
- Ez mindig csak az egyik játékos szempontjait tükrözi.
- Sokszor ez egy *f*:állás → [-100..100] függvény.

Minimax algoritmus

- Adott állásból indulva felépítjük a játékfa néhány szintjét.
- A részfa leveleit kiértékeljük aszerint, hogy azok számunkra kedvező, vagy kedvezőtlen állások.
- Az értékeket felfuttatjuk a fában. (Saját szintjeink csúcsaihoz azok gyermekeinek maximumát, ellenfél csúcsaihoz azok gyermekeinek minimumát rendeljük.)
- Soron következő lépésünk ahhoz az álláshoz vezet, ahonnan a gyökérhez felkerült a legnagyobb érték.

Példa



Megjegyzés

- Az algoritmust minden alkalommal, valahányszor mi következünk, megismételjük, hiszen lehet, hogy az ellenfél nem az általunk várt legerősebb lépésekkel válaszol, mert:
  - eltérő mélységű részfával dolgozik,
  - más kiértékelő függvényt használ,
  - nem minimax eljárást alkalmaz,
  - hibázik.

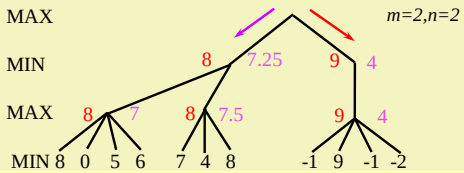
Negamax algoritmus

- Negamax eljárást könnyebb implementálni.
  - Az ellenfél szintjén levő levelek értékének vesszük a (-1)-szeresét, majd
  - minden szinten

    szülő:= max(-gyerek$_1$,..., -gyerek$_n$)

Átlagoló kiértékelés

- Az (*m*,*n*) átlagolás célja a kiértékelő függvény esetleges tévedéseinek simítása.
- A MAX szintjeire a *m* darab legnagyobb, MIN szintjeire az *n* legkisebb értékű gyerek átlaga kerül.



Váltakozó mélységű kiértékelés

- Célja, hogy a kiértékelő függvény minden ágon reális értéket mutasson.
- A részfa felépítését módosítjuk úgy, hogy egy adott szinttől kezdve, akkor vesszük bele egy csúcs utódait a részfába, ha az utódra nem teljesül a nyugalmi teszt:
  - $|f(szülő) - f(gyerek)| < K$

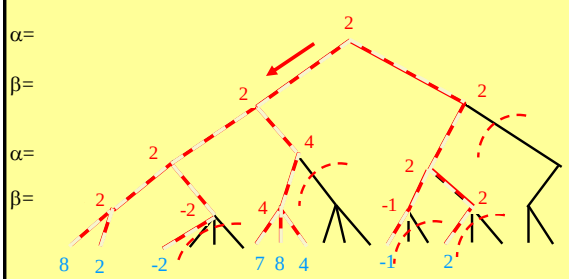
Szelektív kiértékelés

- Célja a memória-igény csökkentése.
- Elkülönítjük a lényeges és lényegtelen lépéseket, és csak a lényeges lépéseknek megfelelő részfát építjük fel.

Alfa-béta algoritmus

- Visszalépéses algoritmus segítségével járjuk be a részfát (mélységi bejárás). Az aktuális úton fekvő csúcsokat:
  - a mi szintünkön α értékkel (ennél rosszabb értékű állásba innen már nem juthatunk),
  - az ellenfelén β értékkel(ennél jobb értékű állásba onnan már nem juthatunk) látjuk el.
- Lefelé haladva $\alpha=-\infty$, és $\beta=+\infty$.
- Ezek visszalépéskor a felhozott értékre változnak, ha az nagyobb, mint az α, illetve kisebb, mint a β.
- Vágás: ha az úton van olyan α és β, hogy $\alpha \geq \beta$.

Példa



Eredmény

- Több egyformán jó kezdőirány esetén a baloldalit kell választani.
- Ekkor ugyanazt a kezdőlépést kapjuk eredményül, mint a minimax algoritmussal talált baloldali legjobb kezdőlépés.

Kétszemélyes játékot játszó program

- Váltakozó mélységű, szelektív, (*m*,*n*) átlagoló, negamax alfa-béta kiértékelést végez.
- Keretprogram, amely váltakozva fogadja a felhasználó lépéseit, és generálja a számítógép lépéseit.
- Kiegészítő funkciók (beállítások, útmutató, segítség, korábbi lépések tárolása, mentés stb.)
- Felhasználói felület, grafika
- Heurisztika megválasztása (kiértékelő függvény, szelekció, kiértékelés sorrendje)