



Projekat 1 - Reversi

Algoritmi i strukture podataka 2020/2021

Na početku

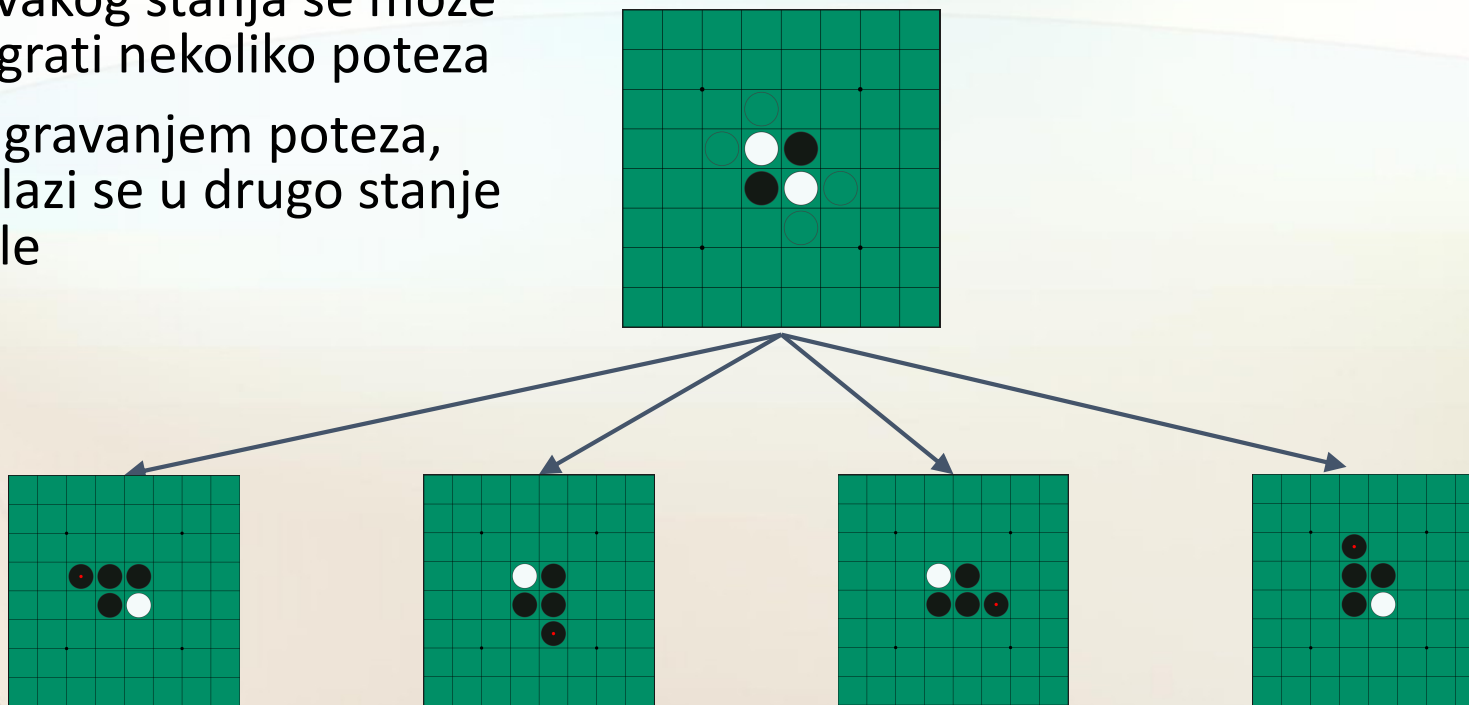
- Odigrajte nekoliko partija igre
- Pokušajte da razvijete/otkrijete dobre taktike

Stanje i modelovanje stanja

- Stanje (table) predstavlja trenutni broj, raspored i vrste figura na tabli
- Potrebno je definisati strukturu podataka koja opisuje trenutno stanje
- Koje strukture bi to mogle biti?

Prelazak u sledeće stanje 1/2

- Iz svakog stanja se može odigrati nekoliko poteza
- Odigravanjem poteza, prelazi se u drugo stanje table

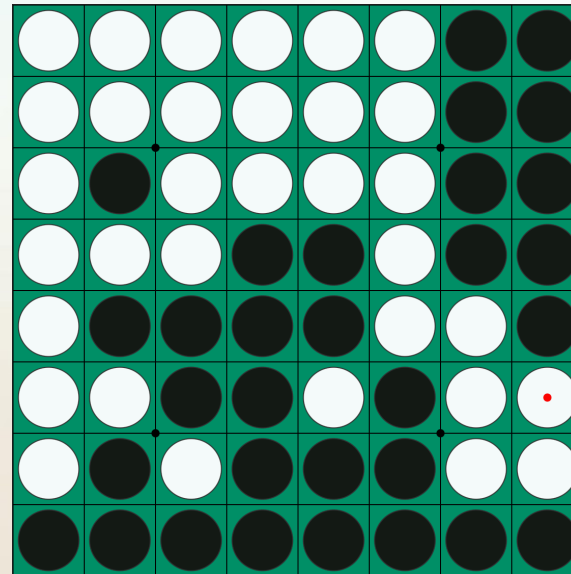
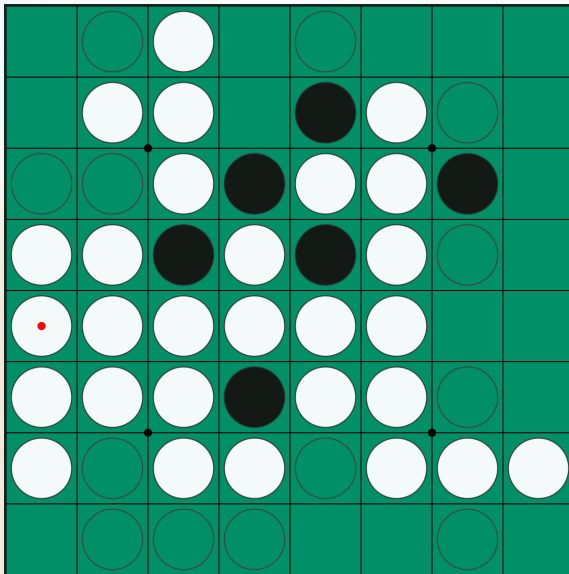


Prelazak u sledeće stanje 2/2

- Nisu svi prelazi jednako povoljni za igrača
- Potrebno je odabrati najbolji od mogućih prelaza
- Stanje sa sledećim mogućim stanjima formira stablo igre
- Kako utvrditi koji od sledećih poteza je najbolji?

Heuristika 1/3

- Da li su ova stanja dobra za belog igrača?



Heuristika 2/3

- Heuristika je funkcija koja određuje kvalitet stanja
- Transformiše stanje table u broj
- Što je broj veći, stanje je poželjnije

Heuristika 3/3

- Šta možemo uključiti u heuristiku?
 - Broj naših figura
 - Broj protivničkih figura
 - Broj naših figura u uglovima (ne mogu biti pojedene)
 - Broj protivničkih figura u uglovima (ne mogu biti pojedene)
 - ...

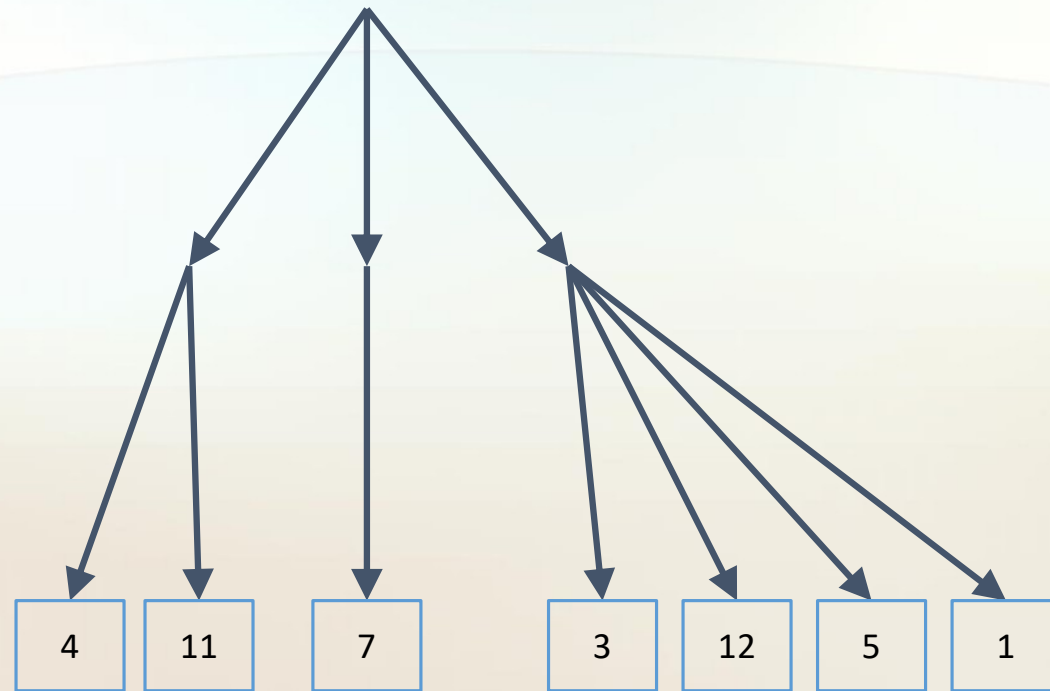
Minimax algoritam

- 2 igrača – svaki želi da pobjedi (postigne što bolji rezultat), a da pritom protivnik izgubi, odnosno postigne najgori mogući rezultat
- Stanje koje je dobro za jednog igrača, obavezno je loše za protivnika
- Jedan igrač – maximizer
- Drugi igrač - minimizer

Minimax algoritam

Maximizer

Minimizer

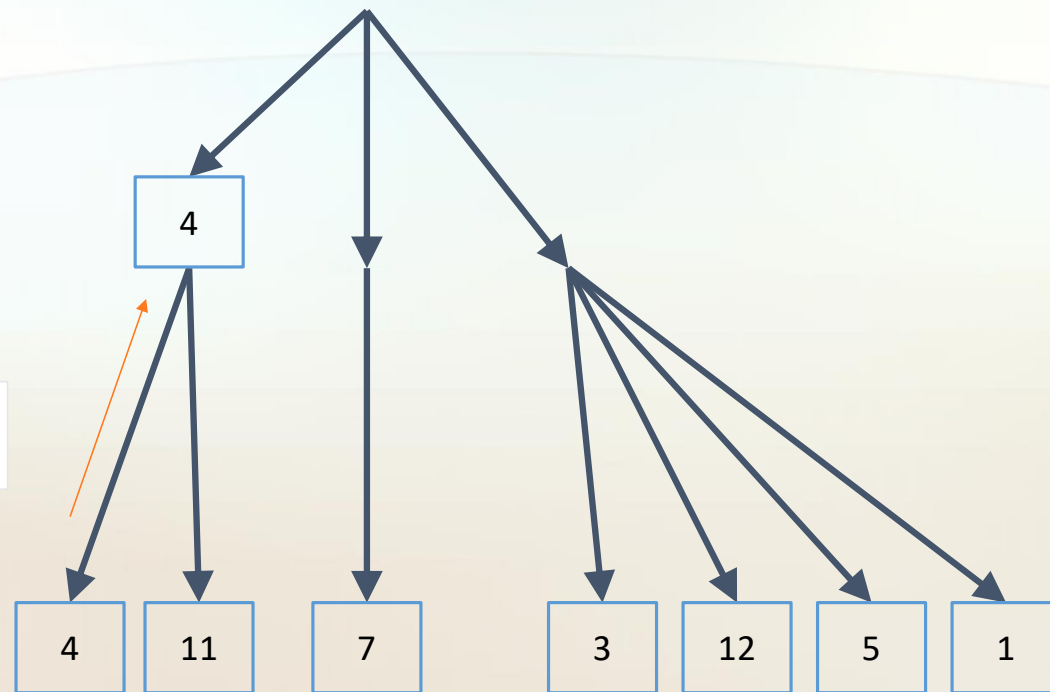


Minimax algoritam

Maximizer

Minimizer

Minimizer bira
manju vrednost

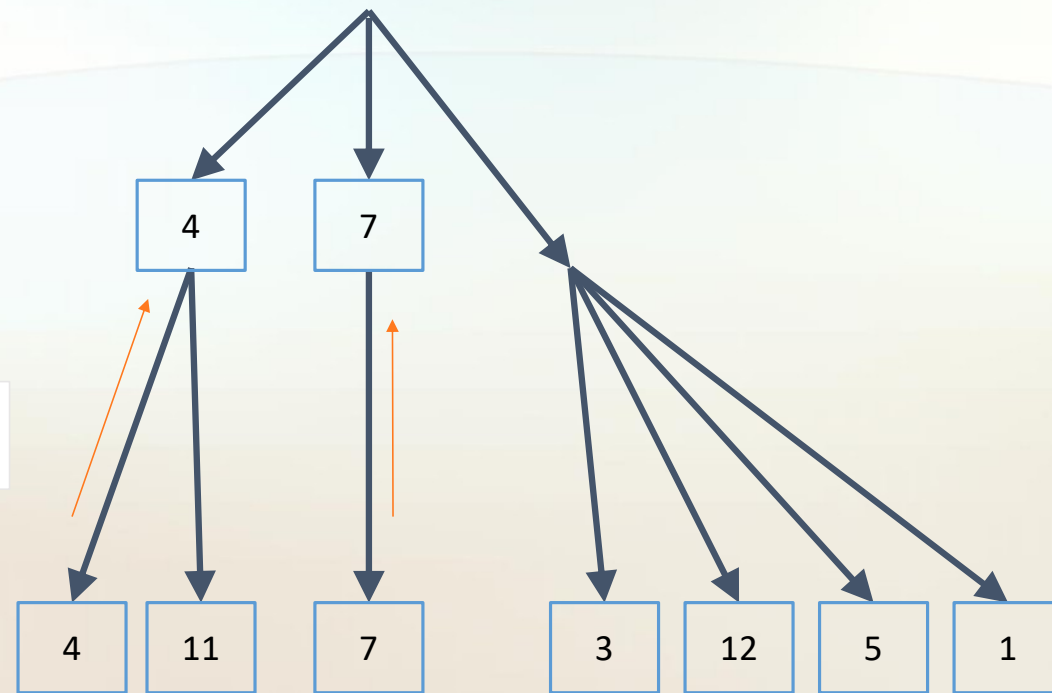


Minimax algoritam

Maximizer

Minimizer

Minimizer bira
manju vrednost

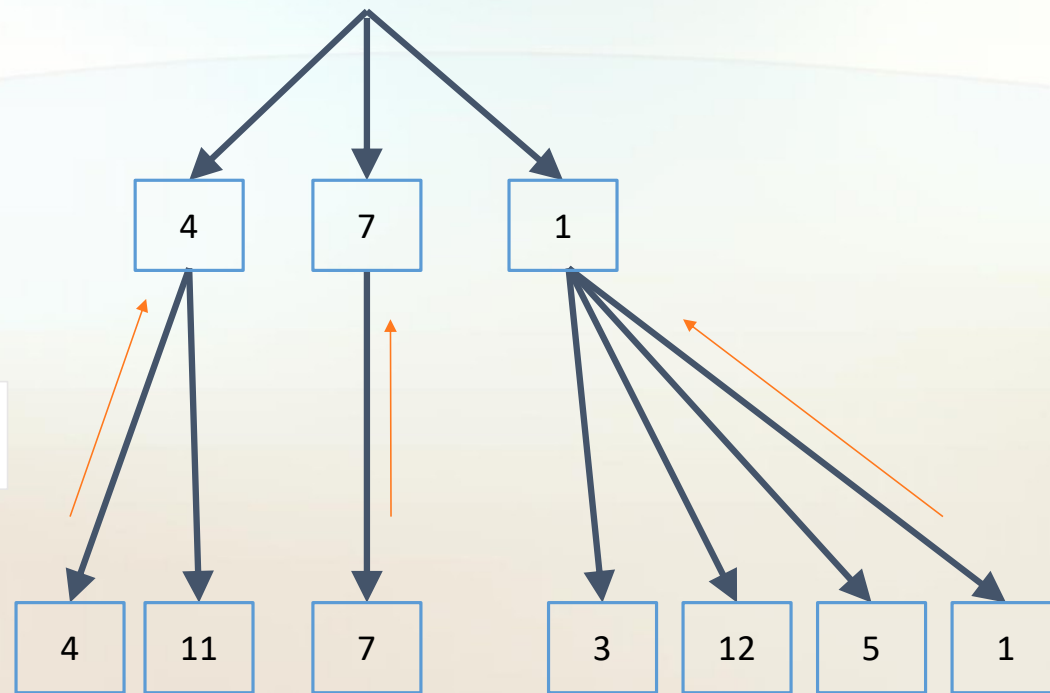


Minimax algoritam

Maximizer

Minimizer

Minimizer bira
manju vrednost

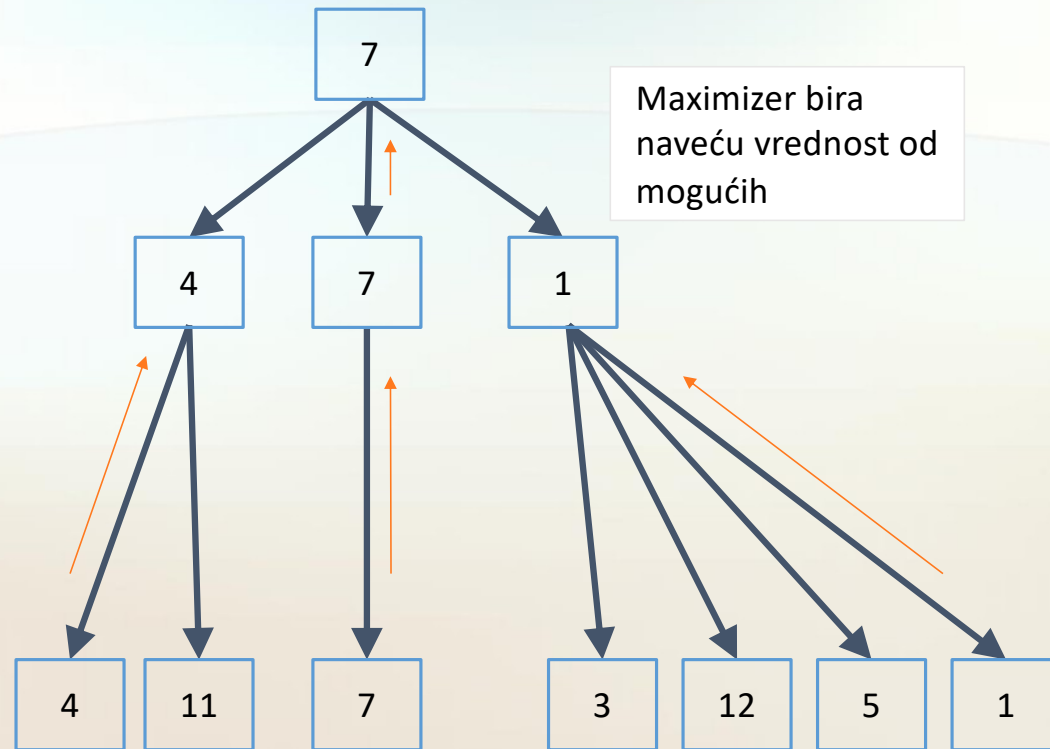


Minimax algoritam

Maximizer

Minimizer

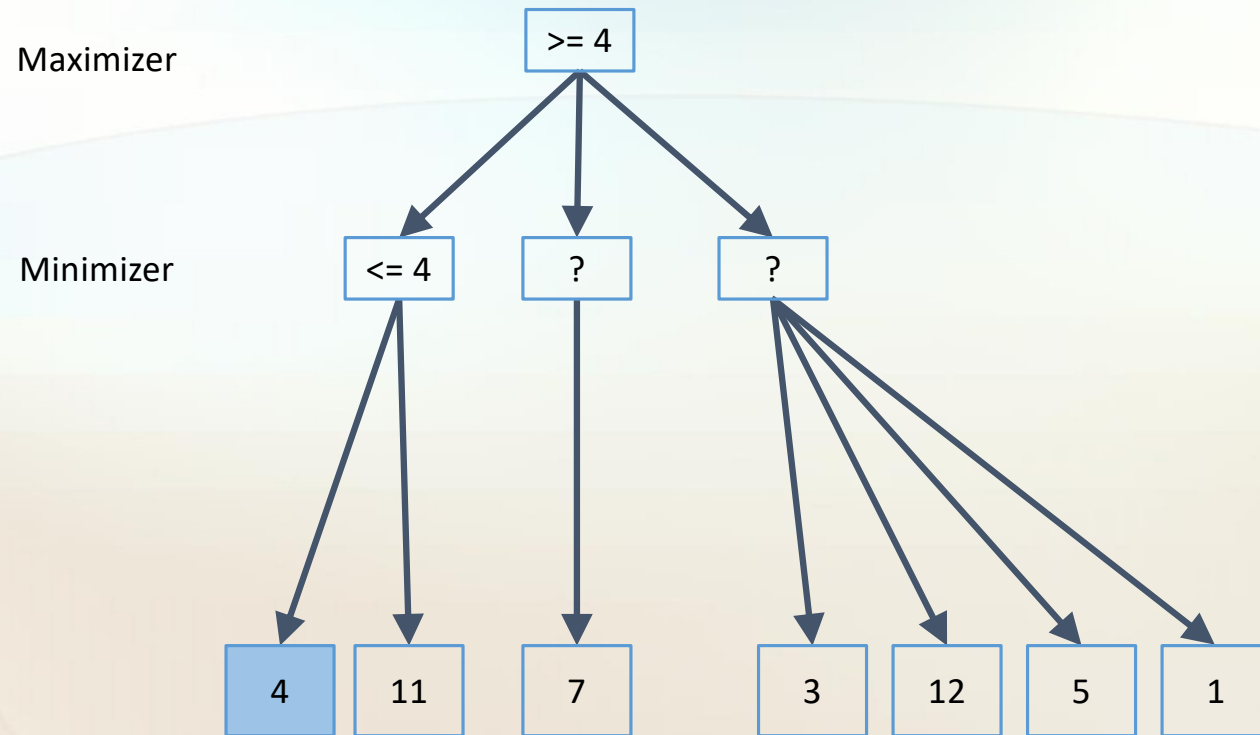
Maximizer bira
naveću vrednost od
mogućih



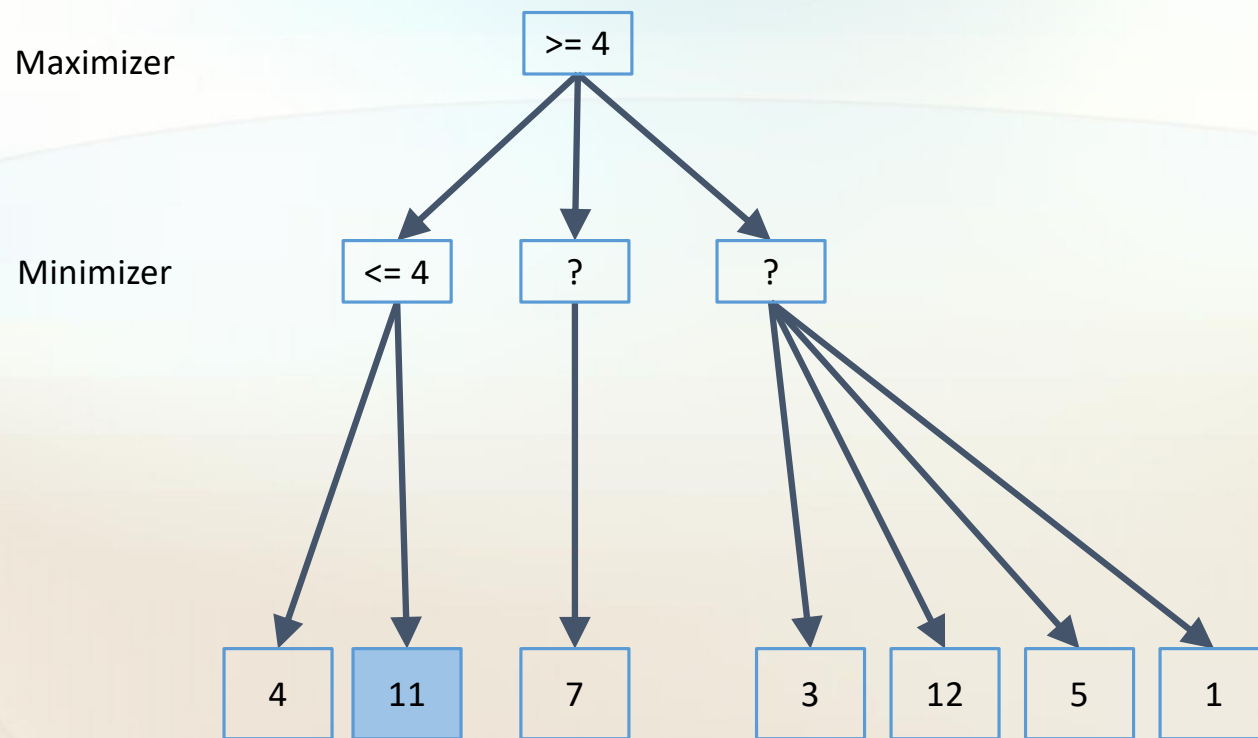
Minimax algoritam sa alfa-beta rezovima

- Nadogradnja minimax algoritma
- Neke putanje možemo preskočiti – ubrzanje
- Uvode se dve konstante – alfa i beta

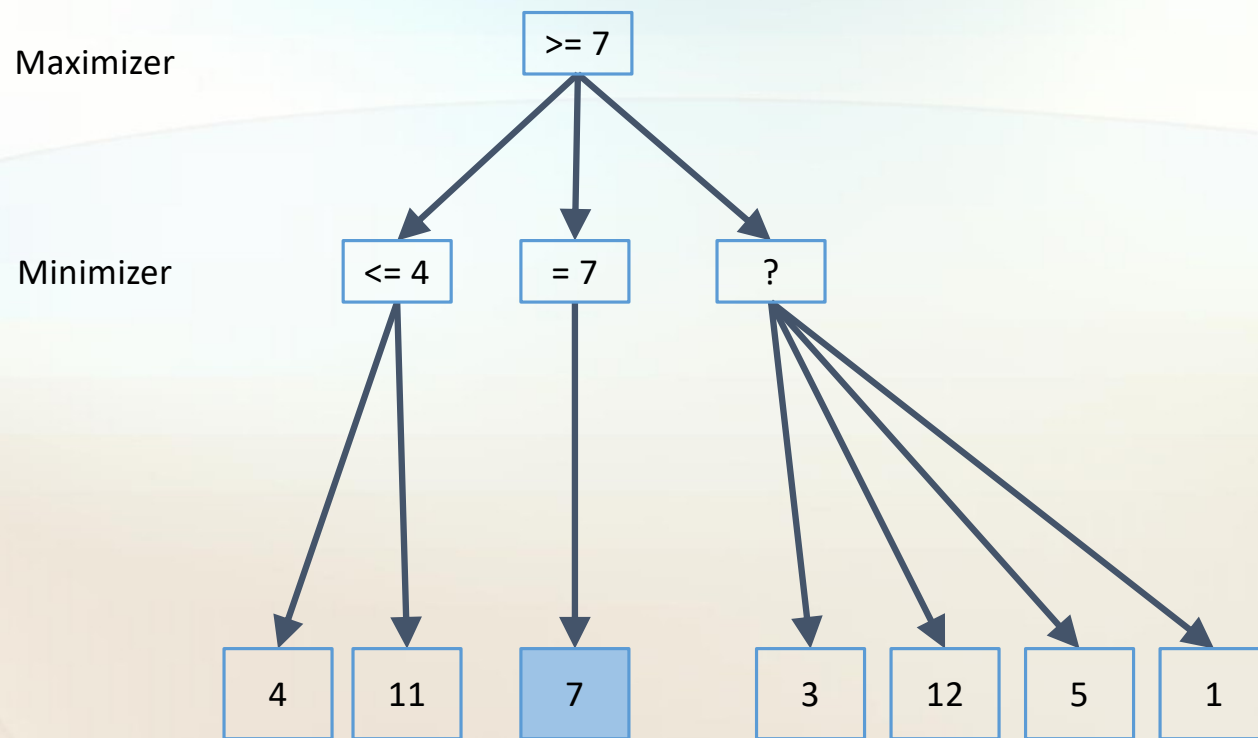
Minimax algoritam sa alfa-beta rezovima



Minimax algoritam sa alfa-beta rezovima



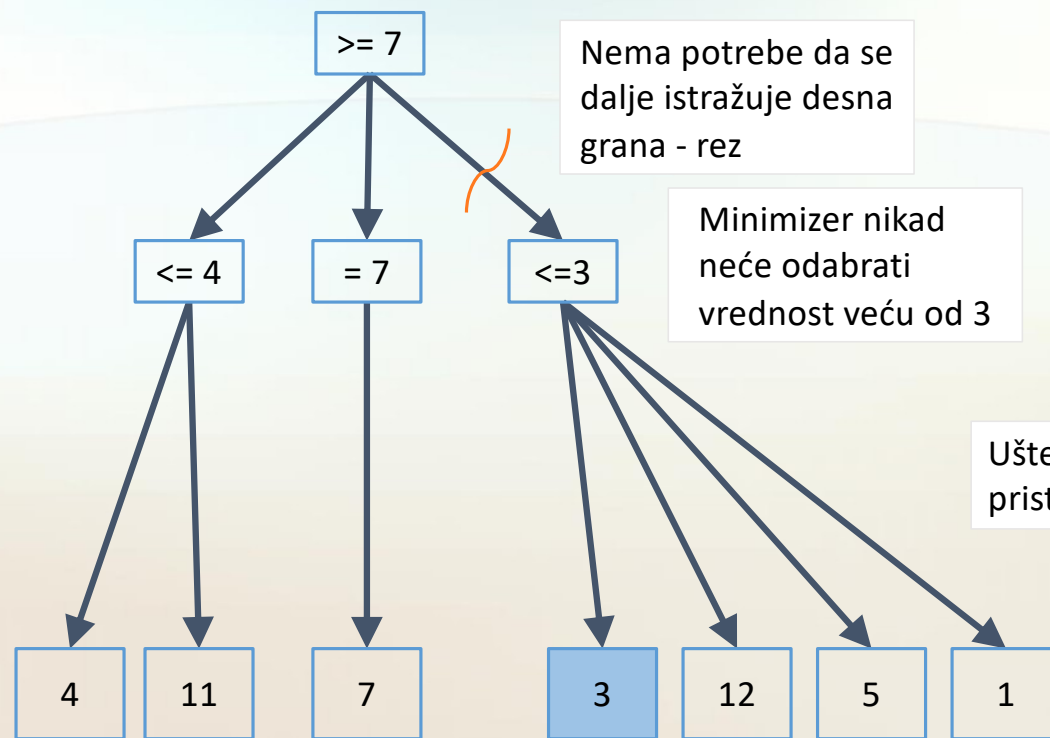
Minimax algoritam sa alfa-beta rezovima



Minimax algoritam sa alfa-beta rezovima

Maximizer

Minimizer



Minimax algoritam sa alfa-beta rezovima

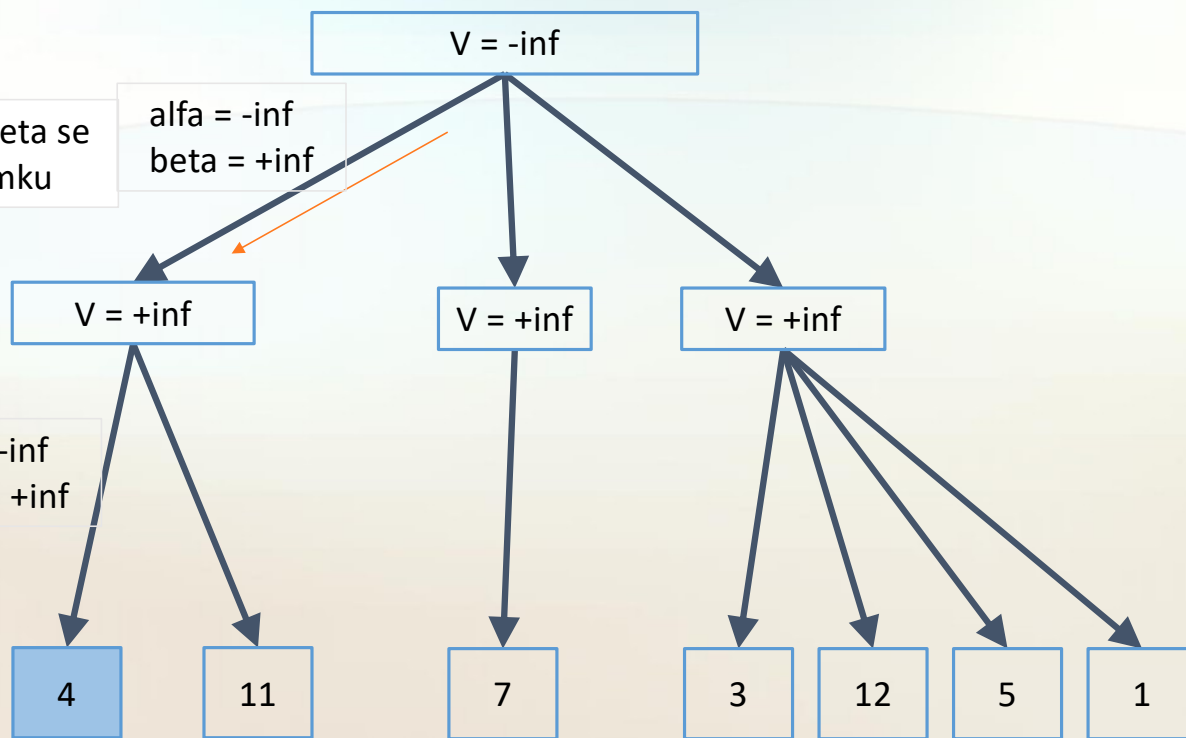
Maximizer

Vrednosti alfa i beta se
prosleđuju potomku

alfa = -inf
beta = +inf

Minimizer

alfa = -inf
beta = +inf



Minimax algoritam sa alfa-beta rezovima

Maximizer

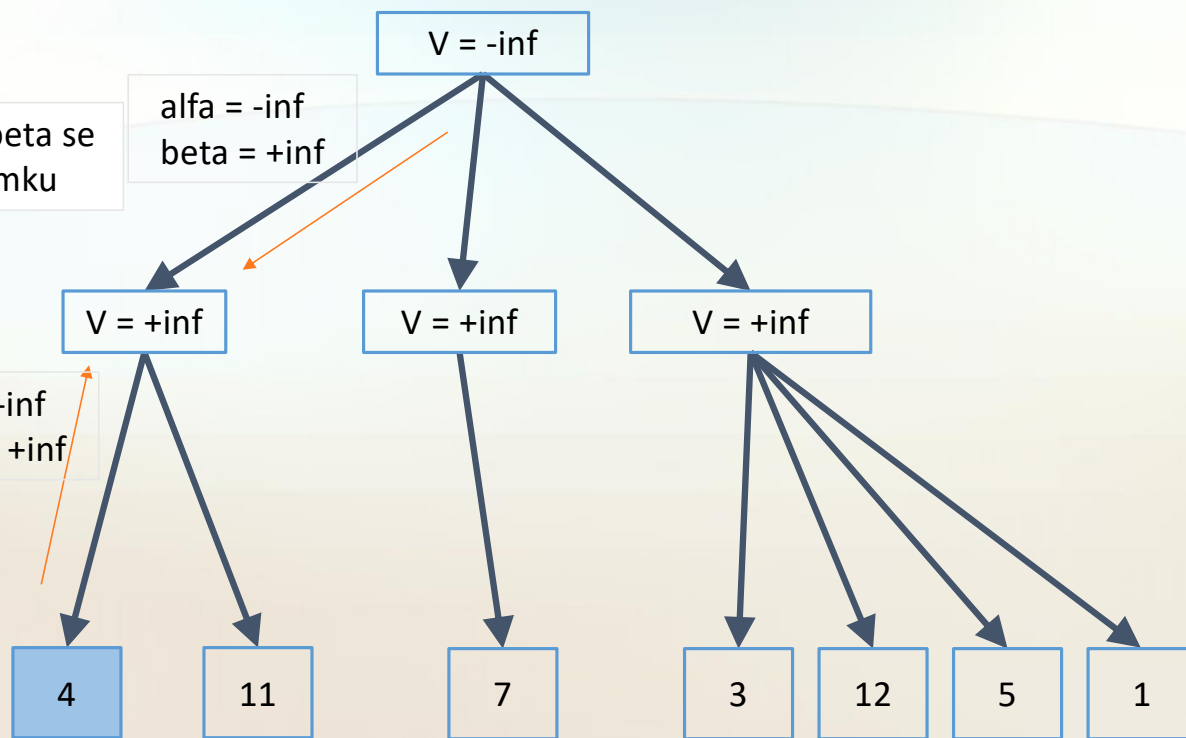
Vrednosti alfa i beta se
prosleđuju potomku

alfa = -inf
beta = +inf

Minimizer

alfa = -inf
beta = +inf

Novopronađena
vrednost 4 se
prosleđuje roditelju
gde se poredi sa
vrednošću i
konstantama alfa i
beta



Minimax algoritam sa alfa-beta rezovima

Maximizer

Vrednosti alfa i beta se prosleđuju potomku

$\alpha = -\text{inf}$
 $\beta = +\text{inf}$

$V = -\text{inf}$

Minimizer

$V = 4$

$V = +\text{inf}$

$V = +\text{inf}$

$\alpha = -\text{inf}$
 $\beta = +\text{inf}$

Novopronađena vrednost 4 se prosleđuje roditelju gde se poredi sa vrednošću i konstantama alfa i beta

Nova vrednost se poredi sa alfa kod minimizer-a

Da je manja, došlo bi do odsecanja grane

4

11

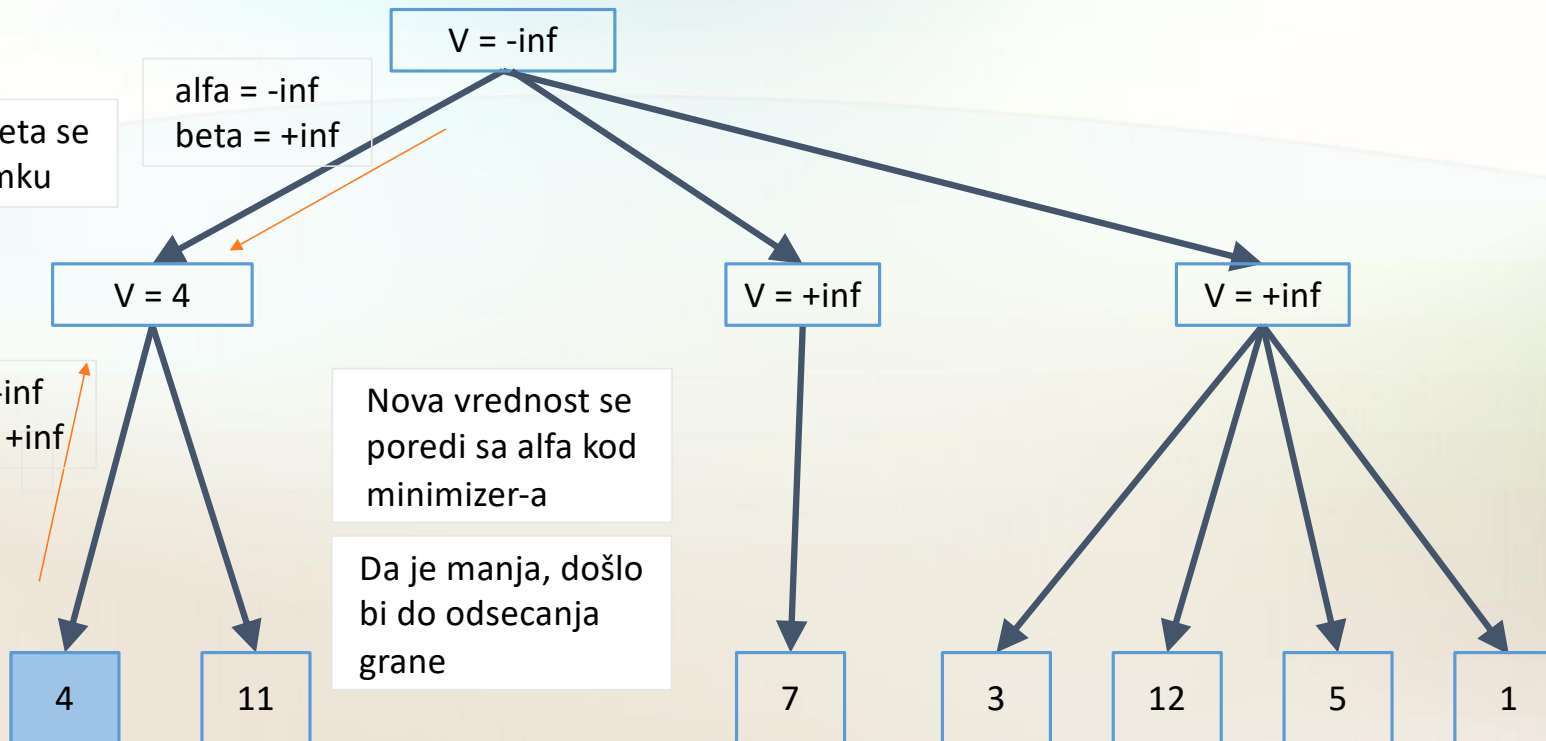
7

3

12

5

1



Minimax algoritam sa alfa-beta rezovima

Maximizer

Vrednosti alfa i beta se prosleđuju potomku

alfa = -inf
beta = +inf

V = -inf

Minimizer

V = 4

V = +inf

V = +inf

Novopronađena vrednost 4 se prosleđuje roditelju gde se poredi sa vrednošću i konstantama alfa i beta

alfa = -inf
beta = +inf

Nova vrednost se poredi sa alfa kod minimizer-a

Da je alfa veća, došlo bi do odsecanja grane

4

11

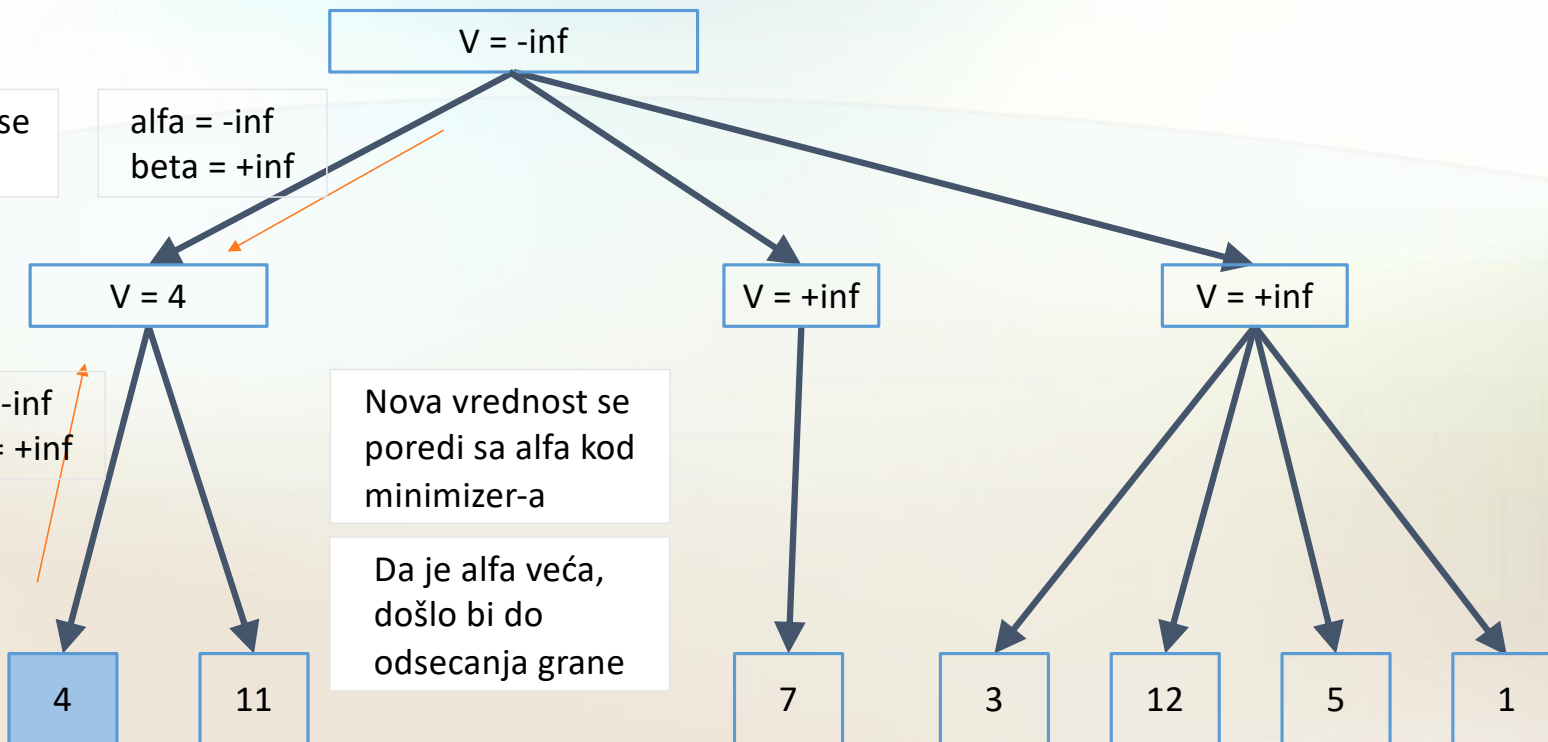
7

3

12

5

1



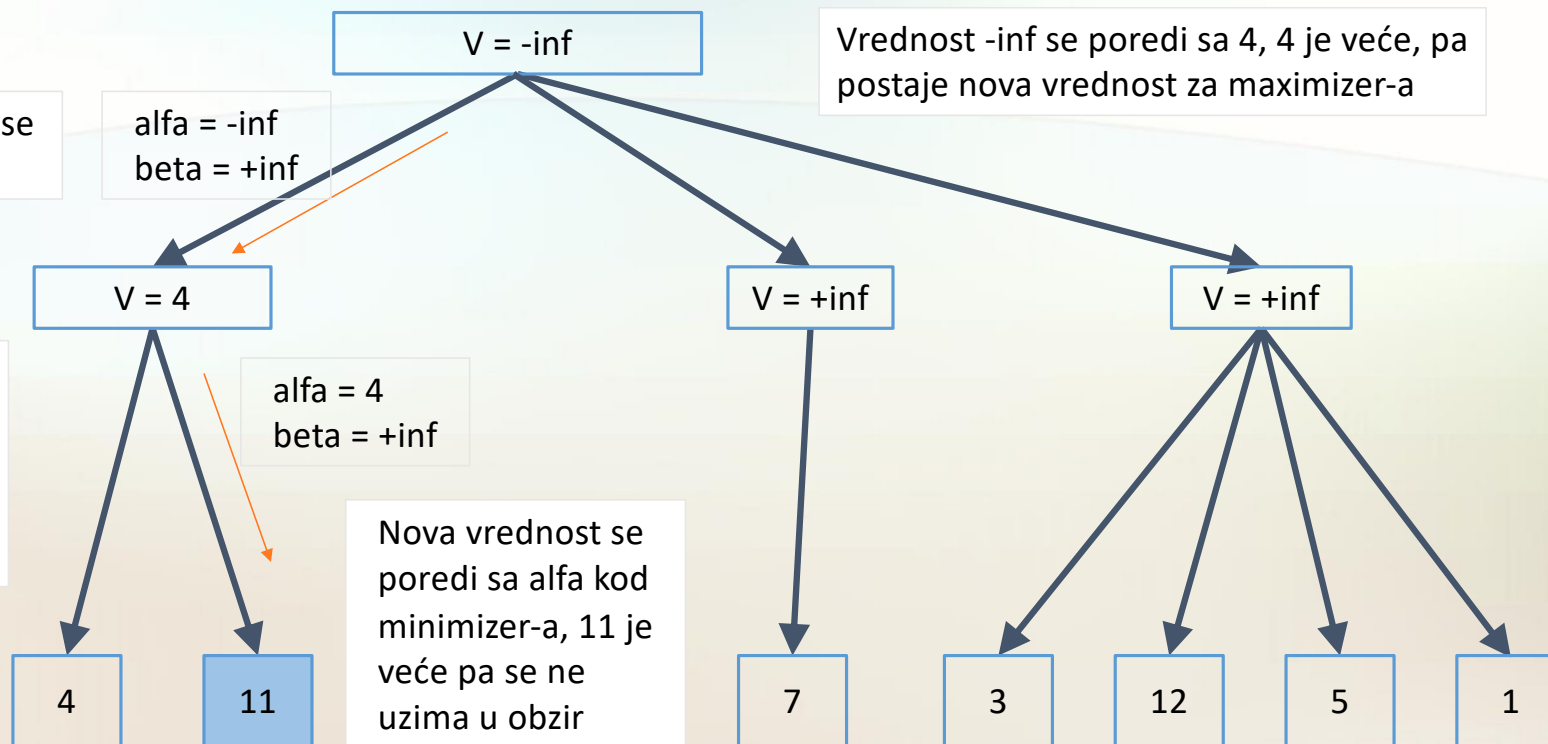
Minimax algoritam sa alfa-beta rezovima

Maximizer

Vrednosti alfa i beta se prosleđuju potomku

Minimizer

Alfa dobija najveću vrednost do korena, Beta dobija najmanju vrednost do korena



Minimax algoritam sa alfa-beta rezovima

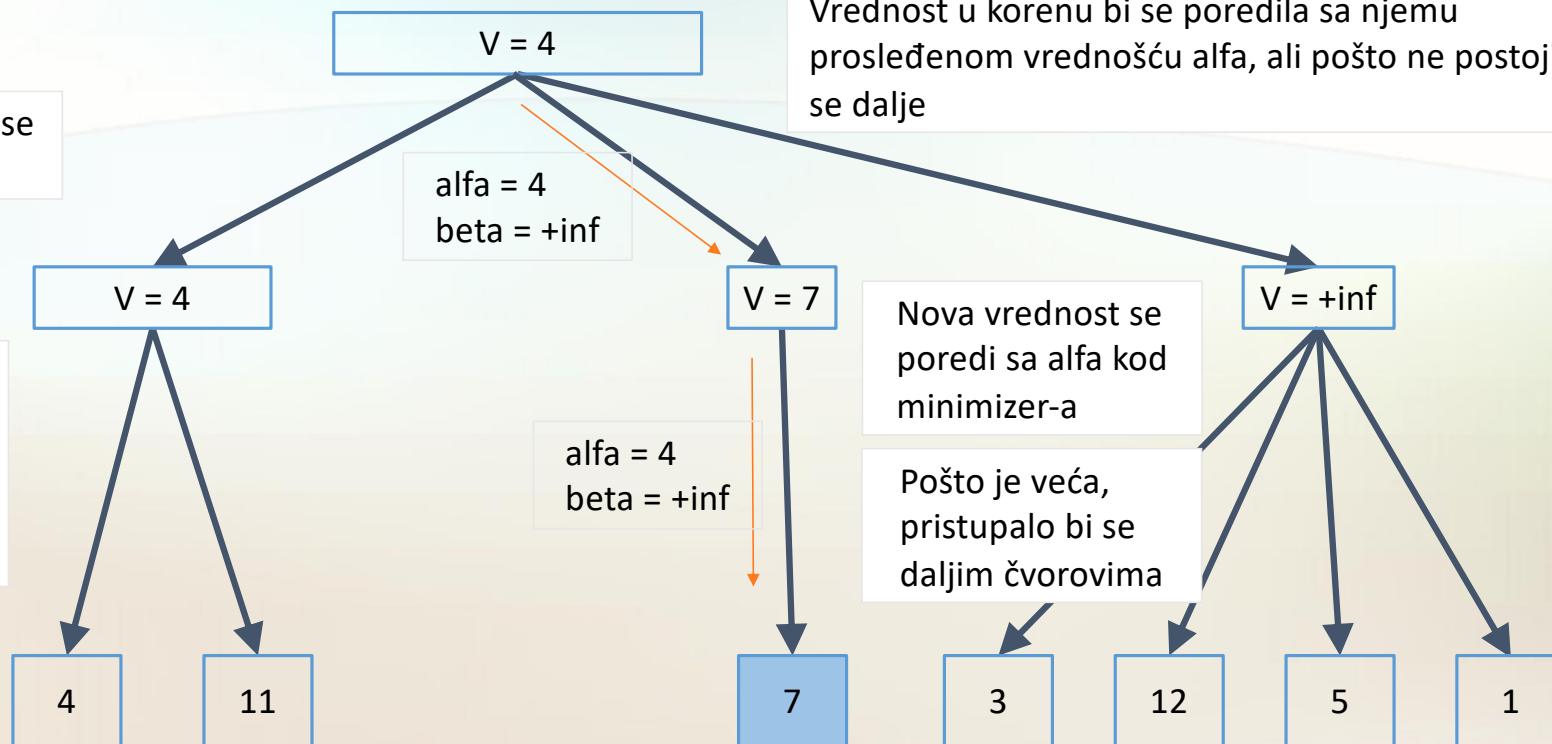
Maximizer

Vrednosti alfa i beta se prosleđuju potomku

Vrednost u korenu bi se poredila sa njemu prosleđenom vrednošću alfa, ali pošto ne postoji, ide se dalje

Minimizer

Alfa dobija najveću vrednost do korena, Beta dobija najmanju vrednost do korena



Minimax algoritam sa alfa-beta rezovima

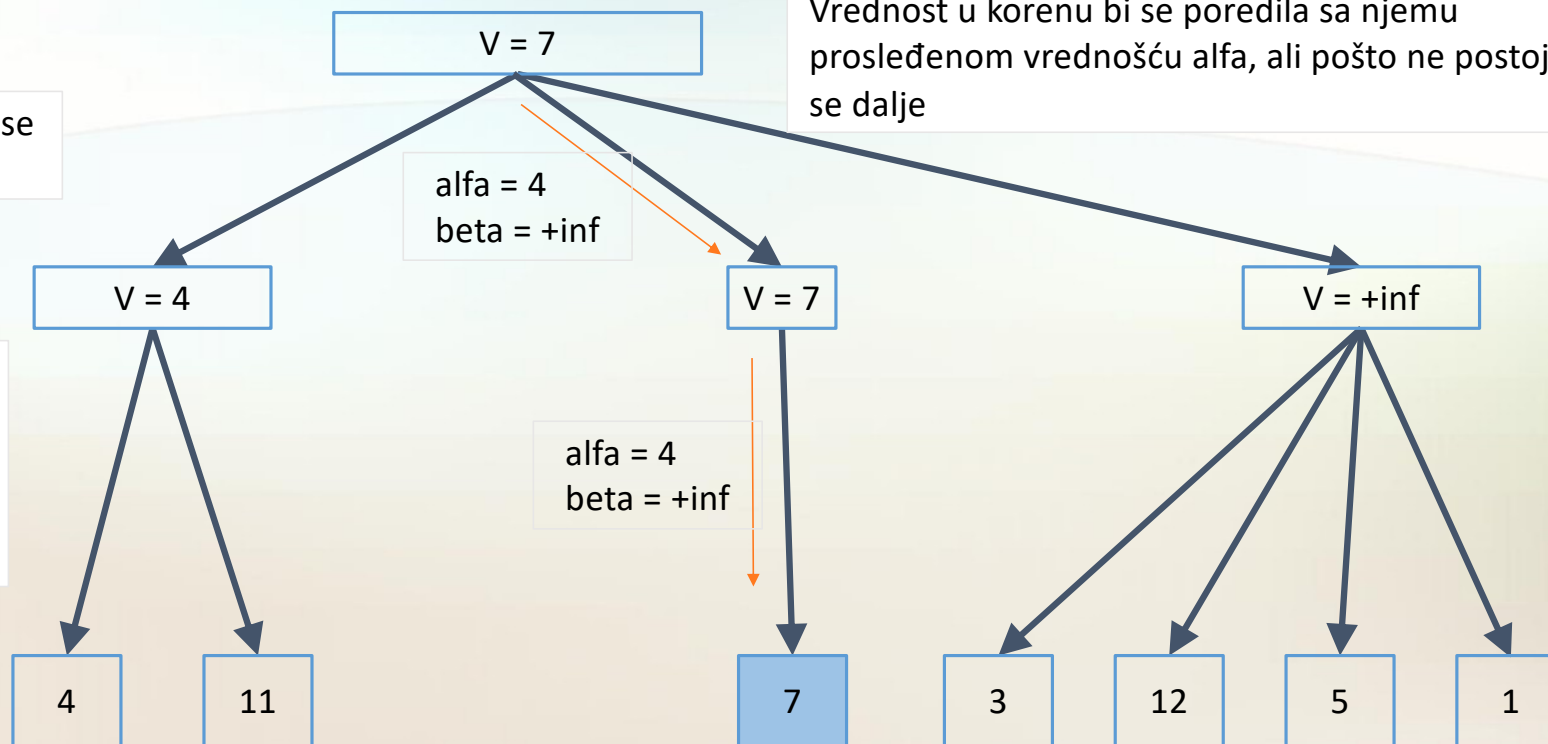
Maximizer

Vrednosti alfa i beta se prosleđuju potomku

Vrednost u korenu bi se poredila sa njemu prosleđenom vrednošću alfa, ali pošto ne postoji, ide se dalje

Minimizer

Alfa dobija najveću vrednost do korena, Beta dobija najmanju vrednost do korena



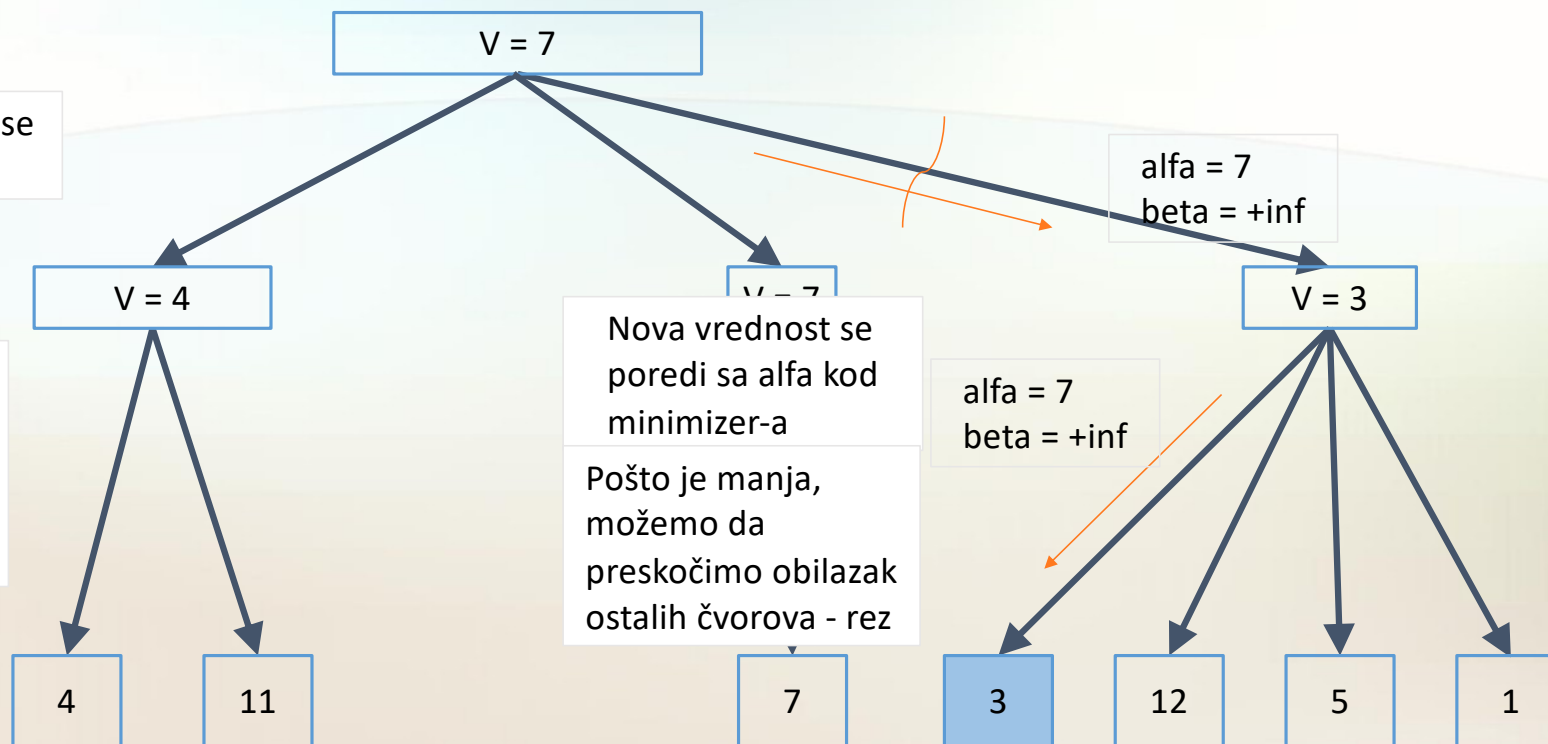
Minimax algoritam sa alfa-beta rezovima

Maximizer

Vrednosti alfa i beta se prosleđuju potomku

Minimizer

Alfa dobija najveću vrednost do korena, Beta dobija najmanju vrednost do korena



Dubina obilaska

- Što veći deo stabla otkrijemo, bolje možemo da odaberemo sledeći potez
- Obilazak većeg dela stabla zahteva više resursa i usporava odlučivanje
- Potrebno je izbalansirati dubinu pretrage sa vremenskim ograničenjem