

# INTELIGENTNI SISTEMI

as. ms Vladimir Jocović  
as. ms Adrian Milaković



# STRATEGIJE REŠAVANJA PROBLEMA

Planiranje

# 05

---

*„A goal without a plan is just a wish.“  
- Antoine de Saint-Exupery*

# PLANIRANJE

## Zašto je planiranje bitno?

Prostor pretraživanja može biti ogroman i korišćenje tradicionalnih algoritama pretrage u procesu pretraživanja ne bi dalo dobre rezultate. Iz tog razloga neophodno je apstrahovati sam problem koji se rešava i dekomponovati ga na najvažnije elemente (potprobleme).

Potproblemi koji rezultuju iz dekompozicije originalnog problema mogu imati međusobnu interakciju. Da bi se razrešio problem međusobne interakcije potproblema neophodno je njihovo planiranje. Stoga je potrebno odrediti redosled izvršavanja akcija, koje se preduzimaju u obradi potproblema, pre nego što se one i same izvrše, što predstavlja proces **planiranja**.

# PLANIRANJE

## Zašto je planiranje bitno?

U dosadašnjim problemima pretrage agentima su bili poznati:

- startno stanje
- jedno ili više ciljnih stanja
- tranziciona tabela, koja određuje stanja u koja je moguće preći

Cilj pretraživanja je pronaći putanju, koja povezuje startno sa jednim od ciljnih stanja.

Sada je poznat i skup akcija koje je moguće preduzeti u svakom stanju, kao i preduslovi i efekti primene takvih akcija.

Sistemi za planiranje zasnivaju se na simulaciji akcija nad bazom stavova, koja predstavlja trenutno stanje posmatranog sveta.

# STRIPS

**ST**anford **R**esearch **I**nstitute **P**roblem **S**olver – sistem za planiranje koji se bazira na logici prvog reda (*First Order Logic*) za predstavljanje stanja sistema i akcija za modifikaciju stanja.

U logici prvog reda svaki svet se sastoji od:

- **Objekata** – u prirodnim jezicima to su imenice, koje referišu objekte (npr. *Ivan, pas, ...*).
- **Relacija** – u prirodnim jezicima to su glagoli koji se odnose na relacije između objekata (npr. *Jaci(Ivan, Milan)*) ili pridevi/prilozi (npr. *Visok(Ivan)*).
- **Funkcija** – relacije u kojima postoji samo jedan rezultat za dati ulaz. Takav izraz referiše jedan objekat. (npr. *Otac(x), Otac(Ivan) == Milan, ...*).

# STRIPS

Svako stanje sistema u STRIPS-u se predstavlja kao konjunkcija prostih stavova. Akcijama se vrši modifikacija trenutnog stanja sistema. Za razliku od logike prvog reda, STRIPS **ne podržava** sledeće elemente za predstavljanje trenutnog stanja:

- **Promenljive** – npr.  $Brat(x, Aca)$ , gde je  $x$  promenljiva.
- **Funkcije** – npr.  $Brat(Otac(Jova), Aca)$ .
- **Univerzalne kvantifikatore** – npr.  $\forall Brat(x, Aca)$ .
- **Egzistencijalne kvantifikatore** – npr.  $\exists Brat(x, Aca)$ .

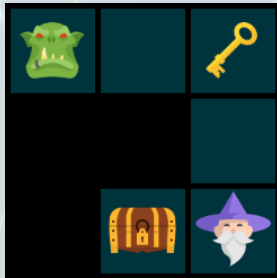
Osnovna ideja STRIPS algoritma nije dokazivanje da neki stav važi (kao u logici prvog reda), već određivanje niza operatora/akcija čijom se primenom dolazi od startnog stanja do željenog stava.



# STRIPS

Osnovni elementi STRIPS algoritma su:

- **Startno (inicijalno) stanje** – stanje od kojeg planer polazi.
- **Ciljno stanje** – stanje koje planer u svom radu pokušava da dosegne.
- **Trenutno stanje** – trenutno stanje sistema predstavljeno konjunkcijom stavova logike prvog reda.



# STRIPS

Osnovni elementi STRIPS algoritma su:

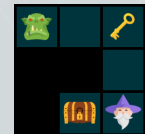
- **Skup akcija** – svaku akciju čine (pored imena):



- **skup preduslova** – konjunkcija stavova koji moraju biti ispunjeni kako bi akcija mogla da se primeni.
- **skup stavova za uklanjanje** – konjunkcija stavova koji će biti uklonjeni iz skupa stavova tekućeg stanja primenom ove akcije.
- **skup stavova za dodavanje** – konjunkcija stavova koji će biti dodati u skup stavova tekućeg stanja primenom ove akcije.

- **Ciljni stek** – sadrži stavove koje treba izvesti da se dosegne ciljni stav.

(ubiti čudovište ← stajati na polju sa čudovištem & imati mač ← uzeti mač iz kovčega ← stajati na polju sa kovčegom & imati ključ ...)





# Zadatak 1 – Problem prenosa blokova



Dati su sledeći predikati za opis stanja blokova na stolu:

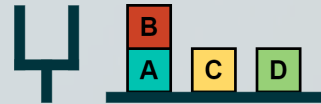
**Na(X, Y)** – Označava da se blok X nalazi neposredno na bloku Y

**NaStolu(X)** – Označava da se blok X nalazi neposredno na stolu

**NaVrhu(X)** – Označava da se iznad bloka X ne nalazi nijedan drugi blok

**RukaPrazna** – Označava da nema blokova u hvataljci

**URuci(X)** – Označava da se u hvataljci nalazi blok X



Date su sledeće akcije koje se mogu primeniti radi izmene stanja:

**UzmiSaStola(X)** – Hvataljkom se uzima blok X koji se nalazi neposredno na stolu i iznad kojeg se ne nalazi nijedan blok.

**SpustiNaSto(X)** – Blok X koji se nalazi u hvataljci se spušta neposredno na sto.

**SkiniSaBlok(X, Y)** – Hvataljkom se uzima blok X koji se nalazi neposredno na bloku Y i iznad kojeg se ne nalazi nijedan blok.

**StaviNaBlok(X, Y)** – Blok X koji se nalazi u hvataljci se spušta neposredno na blok Y iznad kojeg se ne nalazi nijedan blok.

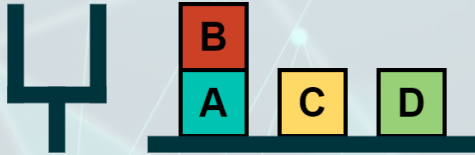
# Zadatak 1 – Problem prenosa blokova



Opisati startno i ciljno stanje prikazano na slici. Definirati potrebne liste stavova (Preduslov, Dodaj, Ukloni) za sve uvedene operatore. Primenom STRIPS algoritma generirati niz akcija kojima bi se skup blokova prebacio iz startnog u ciljno stanje. U hvataljci u jednom trenutku sme da se nađe maksimalno jedan blok. Na stolu može da se nađe proizvoljan broj blokova. Pri izboru akcije prednost dati akciji koja u svojim preduslovima ima više zadovoljenih stavova.



# Zadatak 1 - Rešenje

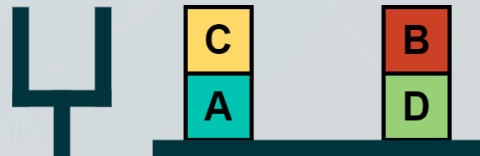


Opis startnog stanja

Na(B, A), NaStolu(A), NaStolu(C), NaStolu(D)  
NaVrhu(B), NaVrhu(C), NaVrhu(D), RukaPrazna

Na(C, A), NaVrhu(C), NaStolu(A)  
Na(B, D), NaVrhu(B), NaStolu(D), RukaPrazna

Opis ciljnog stanja



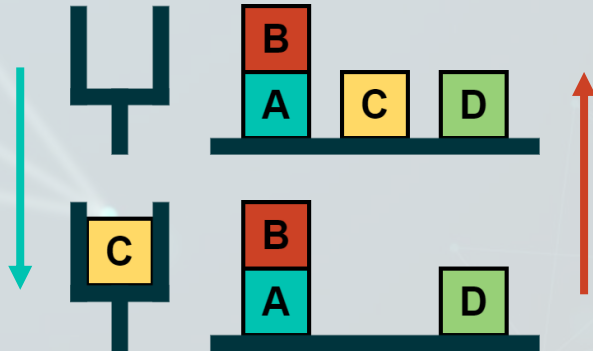
# Zadatak 1 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**



operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

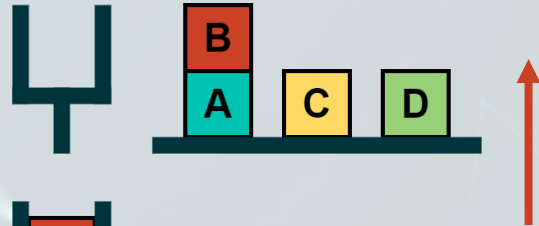
# Zadatak 1 - Rešenje

operator **SkiniSaBloka(X, Y):**

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X) & NaVrhu(Y)**

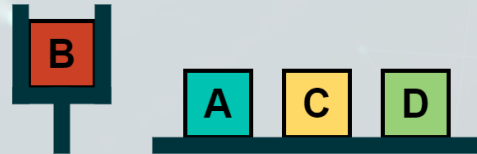


operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

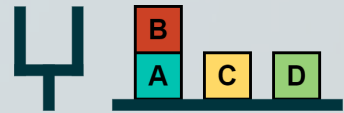
dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**



# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A), \text{NaStolu}(A), \text{NaStolu}(C), \text{NaStolu}(D), \text{NaVrhu}(B),$   
 $\text{NaVrhu}(C), \text{NaVrhu}(D), \text{RukaPrazna}$

goals:



$\text{Na}(C, A) \ \& \ \text{NaVrhu}(C) \ \& \ \text{NaStolu}(A) \ \& \ \text{Na}(B, D) \ \& \ \text{NaVrhu}(B) \ \& \ \text{NaStolu}(D) \ \& \ \text{RukaPrazna}$   
 $\text{Na}(C, A)$   
 $\text{NaVrhu}(C)$   
 $\text{NaStolu}(A)$   
 $\text{Na}(B, D)$   
 $\text{NaVrhu}(B)$   
 $\text{NaStolu}(D)$   
 $\text{RukaPrazna}$

actions:

-

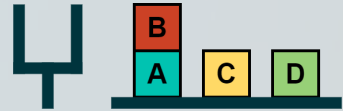
plan:

-



# Zadatak 1 - Rešenje

state: Na(B, A), NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(B),  
NaVrhu(C), NaVrhu(D), RukaPrazna



goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NaVrhu(B)  
NaStolu(D)  
RukaPrazna

actions:

-

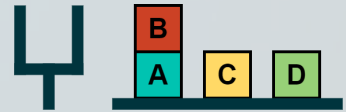
plan:

-

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(B)$ ,  
 $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(D)$ ,  $\text{RukaPrazna}$

goals:



$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

actions:

-

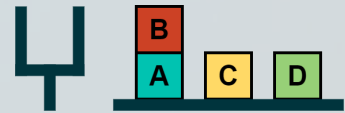
plan:

-

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(B)$ ,  
 $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(D)$ ,  $\text{RukaPrazna}$

goals:



$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

actions:

-

plan:

-

# Zadatak 1 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

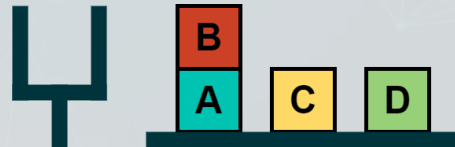
operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

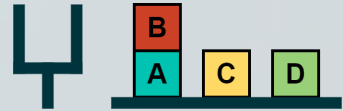
Na(B, D)?



# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(B)$ ,  $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(D)$ ,  $\text{RukaPrazna}$

goals:



$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B)$  &  $\text{NaVrhu}(D)$

$\text{URuci}(B)$

$\text{NaVrhu}(D)$

actions:

-

plan:

-

operator  $\text{StaviNaBlok}(X, Y)$ :

preduslov:  $\text{URuci}(X)$  &  $\text{NaVrhu}(Y)$

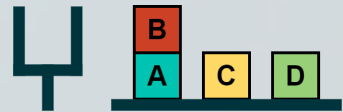
ukloni:  $\text{URuci}(X)$  &  $\text{NaVrhu}(Y)$

dodaj:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{Na}(X, Y)$

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A), \text{NaStolu}(A), \text{NaStolu}(C), \text{NaStolu}(D), \text{NaVrhu}(B),$   
 $\text{NaVrhu}(C), \text{NaVrhu}(D), \text{RukaPrazna}$

goals:



$\text{Na}(C, A) \& \text{NaVrhu}(C) \& \text{NaStolu}(A) \& \text{Na}(B, D) \& \text{NaVrhu}(B) \& \text{NaStolu}(D) \& \text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B) \& \text{NaVrhu}(D)$

$\text{URuci}(B)$

$\text{NaVrhu}(D)$

actions:

$\text{StaviNaBlok}(B, D)$

plan:

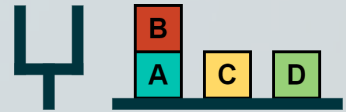
-



# Zadatak 1 - Rešenje

state: Na(B, A), NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(B),  
NaVrhu(C), **NaVrhu(D)**, RukaPrazna

goals:



Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NaVrhu(C)

NaStolu(A)

Na(B, D)

NONE

URuci(B) & NaVrhu(D)

URuci(B)

**NaVrhu(D)**

actions:

**StaviNaBlok(B, D)**

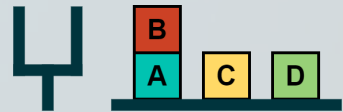
plan:

-

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A), \text{NaStolu}(A), \text{NaStolu}(C), \text{NaStolu}(D), \text{NaVrhu}(B),$   
 $\text{NaVrhu}(C), \text{NaVrhu}(D), \text{RukaPrazna}$

goals:



$\text{Na}(C, A) \& \text{NaVrhu}(C) \& \text{NaStolu}(A) \& \text{Na}(B, D) \& \text{NaVrhu}(B) \& \text{NaStolu}(D) \& \text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B) \& \text{NaVrhu}(D)$

$\text{URuci}(B)$

actions:

$\text{StaviNaBlok}(B, D)$

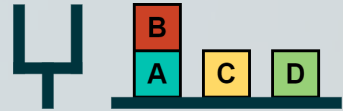
plan:

-

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A), \text{NaStolu}(A), \text{NaStolu}(C), \text{NaStolu}(D), \text{NaVrhu}(B),$   
 $\text{NaVrhu}(C), \text{NaVrhu}(D), \text{RukaPrazna}$

goals:



$\text{Na}(C, A) \ \& \ \text{NaVrhu}(C) \ \& \ \text{NaStolu}(A) \ \& \ \text{Na}(B, D) \ \& \ \text{NaVrhu}(B) \ \& \ \text{NaStolu}(D) \ \& \ \text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B) \ \& \ \text{NaVrhu}(D)$

$\text{URuci}(B)$

actions:

$\text{StaviNaBlok}(B, D)$

plan:

-

# Zadatak 1 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna** & **NaVrhu(X)** & **NaStolu(X)**

ukloni: **RukaPrazna** & **NaVrhu(X)** & **NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna** & **NaVrhu(X)** & **NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna** & **NaVrhu(X)** & **Na(X, Y)**

ukloni: **RukaPrazna** & **NaVrhu(X)** & **Na(X, Y)**

dodaj: **URuci(X)**, **NaVrhu(Y)**

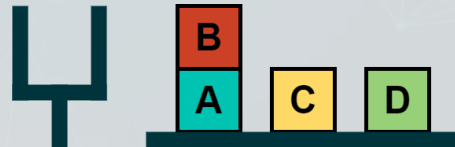
operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X)**, **NaVrhu(Y)**

ukloni: **URuci(X)**, **NaVrhu(Y)**

dodaj: **RukaPrazna** & **NaVrhu(X)** & **Na(X, Y)**

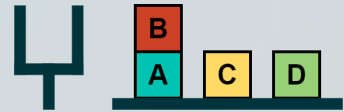
**URuci(B)?**



# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(B)$ ,  
 $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(D)$ ,  $\text{RukaPrazna}$

goals:



$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B)$  &  $\text{NaVrhu}(D)$

$\text{URuci}(B)$

NONE

$\text{RukaPrazna}$  &  $\text{NaVrhu}(B)$  &  $\text{Na}(B, A)$

$\text{RukaPrazna}$

$\text{NaVrhu}(B)$

$\text{Na}(B, A)$

actions:

**StaviNaBlok(B, D)**

plan:

-

operator **SkiniSaBloka(X, Y):**

preduslov:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{Na}(X, Y)$

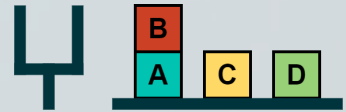
ukloni:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{Na}(X, Y)$

dodaj:  $\text{URuci}(X)$  &  $\text{NaVrhu}(Y)$

# Zadatak 1 - Rešenje

state: Na(B, A), NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(B),  
NaVrhu(C), NaVrhu(D), RukaPrazna

goals:



Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NaVrhu(C)

NaStolu(A)

Na(B, D)

NONE

URuci(B) & NaVrhu(D)

URuci(B)

NONE

RukaPrazna & NaVrhu(B) & Na(B, A)

RukaPrazna

NaVrhu(B)

Na(B, A)

actions:

StaviNaBlok(B, D)

SkiniSaBloka(B, A)

plan:

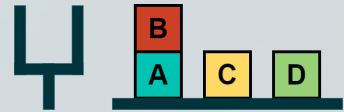
-



# Zadatak 1 - Rešenje

state: **Na(B, A)**, NaStolu(A), NaStolu(C), NaStolu(D), **NaVrhu(B)**,  
NaVrhu(C), NaVrhu(D), **RukaPrazna**

goals:



**Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna**

**Na(C, A)**

**NaVrhu(C)**

**NaStolu(A)**

**Na(B, D)**

**NONE**

**URuci(B) & NaVrhu(D)**

**URuci(B)**

**NONE**

**RukaPrazna & NaVrhu(B) & Na(B, A)**

**RukaPrazna**

**NaVrhu(B)**

**Na(B, A)**

actions:

**StaviNaBlok(B, D)**

**SkiniSaBloka(B, A)**

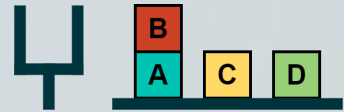
plan:

-

# Zadatak 1 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(B)$ ,  $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(D)$ ,  $\text{RukaPrazna}$

goals:



$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

$\text{NaVrhu}(C)$

$\text{NaStolu}(A)$

$\text{Na}(B, D)$

NONE

$\text{URuci}(B)$  &  $\text{NaVrhu}(D)$

$\text{URuci}(B)$

NONE

actions:

$\text{StaviNaBlok}(B, D)$

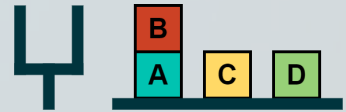
$\text{SkinisaBloka}(B, A)$

plan:

-

# Zadatak 1 - Rešenje

state: Na(B, A), NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(B),  
NaVrhu(C), NaVrhu(D), RukaPrazna



goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NaVrhu(C)

NaStolu(A)

Na(B, D)

NONE

URuci(B) & NaVrhu(D)

URuci(B)

**NONE**

actions:

**StaviNaBlok(B, D)**

**SkiniSaBloka(B, A)**

plan:

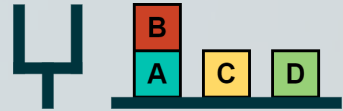
-

# Zadatak 1 - Rešenje

state: **Na(B, A)**, NaStolu(A), NaStolu(C), NaStolu(D), **NaVrhu(B)**,  
NaVrhu(C), NaVrhu(D), **RukaPrazna**

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NONE  
URuci(B) & NaVrhu(D)  
URuci(B)



actions: **StaviNaBlok(B, D)** plan: **SkiniSaBloka(B, A)**

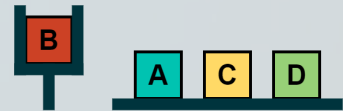
operator **SkiniSaBloka(X, Y)**:  
preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
dodaj: **URuci(X) & NaVrhu(Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(D),  
**URuci(B), NaVrhu(A)**

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NONE  
URuci(B) & NaVrhu(D)  
URuci(B)



actions: plan:  
**StaviNaBlok(B, D)** **SkiniSaBloka(B, A)**

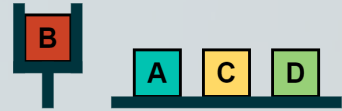
operator **SkiniSaBloka(X, Y):**  
preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
dodaj: **URuci(X) & NaVrhu(Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(D),  
URuci(B), NaVrhu(A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NONE  
URuci(B) & NaVrhu(D)  
URuci(B)



actions:

StaviNaBlok(B, D)

plan:

SkiniSaBloka(B, A)

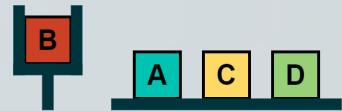


# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(D),  
**URuci(B)**, NaVrhu(A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NONE  
**URuci(B) & NaVrhu(D)**  
**URuci(B)**



actions:

StaviNaBlok(B, D)

plan:

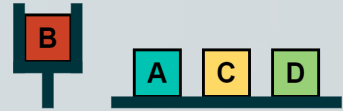
SkiniSaBloka(B, A)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(D),  
URuci(B), NaVrhu(A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
NONE



actions:

StaviNaBlok(B, D)

plan:

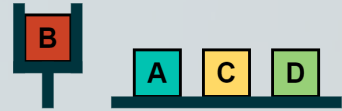
SkiniSaBloka(B, A)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(D),  
URuci(B), NaVrhu(A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)  
**NONE**



actions:

**StaviNaBlok(B, D)**

plan:

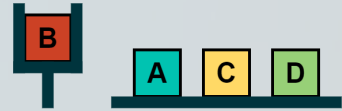
**SkiniSaBloka(B, A)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), **NaVrhu(D)**,  
**URuci(B)**, NaVrhu(A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)



actions:

-

plan:

SkiniSaBlok(B, A)  
**StaviNaBlok(B, D)**

operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

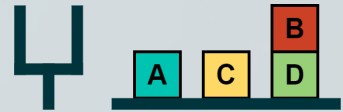
dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)



actions:

-

plan:

SkiniSaBlok(B, A)  
StaviNaBlok(B, D)

operator **StaviNaBlok(X, Y):**

preduslov: URuci(X) & NaVrhu(Y)

ukloni: URuci(X) & NaVrhu(Y)

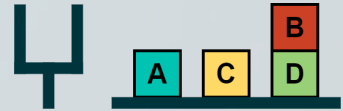
dodaj: RukaPrazna & NaVrhu(X) & Na(X, Y)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NaVrhu(C)  
NaStolu(A)  
Na(B, D)



actions:

-

plan:

SkiniSaBlok(B, A)  
StaviNaBlok(B, D)



# Zadatak 1 - Rešenje

state: **NaStolu(A)**, NaStolu(C), NaStolu(D), **NaVrhu(C)**, NaVrhu(A),  
RukaPrazna, NaVrhu(B), **Na(B, D)**

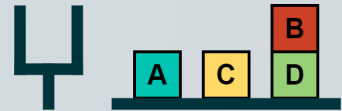
goals:

**Na(C, A)** & **NaVrhu(C)** & **NaStolu(A)** & **Na(B, D)** & **NaVrhu(B)** & **NaStolu(D)** & **RukaPrazna**  
**Na(C, A)**

**NaVrhu(C)**

**NaStolu(A)**

**Na(B, D)**



actions:

-

plan:

**SkiniSaBlok(B, A)**

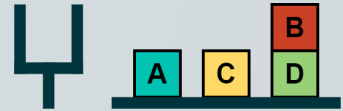
**StaviNaBlok(B, D)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)



actions:

-

plan:

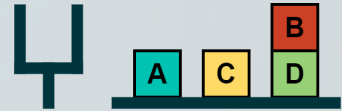
SkiniSaBlok(B, A)  
StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A), RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)



actions:

-

plan:

SkiniSaBlok(B, A)  
StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

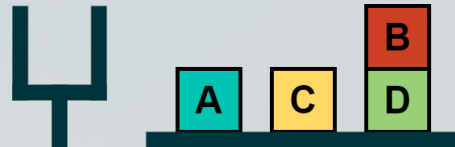
operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

Na(C, A)?



# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

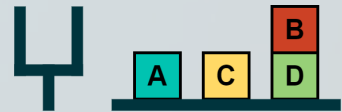
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

NaVrhu(A)



actions:

-

plan:

SkiniSaBlok(B, A)

StaviNaBlok(B, D)

operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

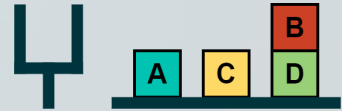
Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

NaVrhu(A)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)



# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), **NaVrhu(A)**,  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

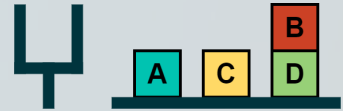
Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

**NaVrhu(A)**



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

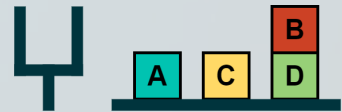
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

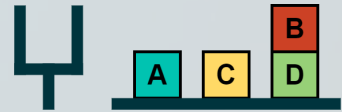
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

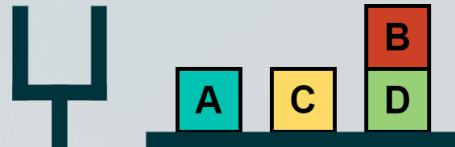
operator **UzmiSaStola(X)**:  
preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**  
ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**  
dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:  
preduslov: **URuci(X)**  
ukloni: **URuci(X)**  
dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:  
preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**  
dodaj: **URuci(X), NaVrhu(Y)**

operator **StaviNaBlok(X, Y)**:  
preduslov: **URuci(X) & NaVrhu(Y)**  
ukloni: **URuci(X) & NaVrhu(Y)**  
dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

URuci(C)?



# Zadatak 1 - Rešenje

state:  $\text{NaStolu}(A)$ ,  $\text{NaStolu}(C)$ ,  $\text{NaStolu}(D)$ ,  $\text{NaVrhu}(C)$ ,  $\text{NaVrhu}(A)$ ,  
 $\text{RukaPrazna}$ ,  $\text{NaVrhu}(B)$ ,  $\text{Na}(B, D)$

goals:

$\text{Na}(C, A)$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(A)$  &  $\text{Na}(B, D)$  &  $\text{NaVrhu}(B)$  &  $\text{NaStolu}(D)$  &  $\text{RukaPrazna}$

$\text{Na}(C, A)$

NONE

$\text{URuci}(C)$  &  $\text{NaVrhu}(A)$

$\text{URuci}(C)$

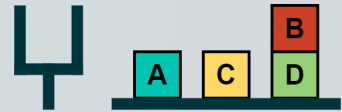
NONE

$\text{RukaPrazna}$  &  $\text{NaVrhu}(C)$  &  $\text{NaStolu}(C)$

$\text{RukaPrazna}$

$\text{NaVrhu}(C)$

$\text{NaStolu}(C)$



actions:

$\text{StaviNaBlok}(C, A)$

plan:

$\text{SkinisaBloka}(B, A)$

$\text{StaviNaBlok}(B, D)$

operator  $\text{UzmiSaStola}(X)$ :

preduslov:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{NaStolu}(X)$

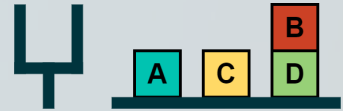
ukloni:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{NaStolu}(X)$

dodaj:  $\text{URuci}(X)$

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:



Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

NONE

RukaPrazna & NaVrhu(C) & NaStolu(C)

RukaPrazna

NaVrhu(C)

NaStolu(C)

actions:

StaviNaBlok(C, A)

UzmiSaStola(C)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)



# Zadatak 1 - Rešenje

state: **NaStolu(A)**, **NaStolu(C)**, NaStolu(D), **NaVrhu(C)**, NaVrhu(A), **RukaPrazna**, NaVrhu(B), Na(B, D)

goals:

**Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna**

**Na(C, A)**

**NONE**

**URuci(C) & NaVrhu(A)**

**URuci(C)**

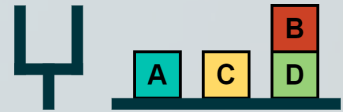
**NONE**

**RukaPrazna & NaVrhu(C) & NaStolu(C)**

**RukaPrazna**

**NaVrhu(C)**

**NaStolu(C)**



actions:

**StaviNaBlok(C, A)**

**UzmiSaStola(C)**

plan:

**SkiniSaBloka(B, A)**

**StaviNaBlok(B, D)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A),  
RukaPrazna, NaVrhu(B), Na(B, D)

goals:

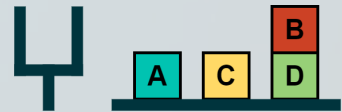
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

NONE



actions:

StaviNaBlok(C, A)

UzmiSaStola(C)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(C), NaStolu(D), NaVrhu(C), NaVrhu(A), RukaPrazna, NaVrhu(B), Na(B, D)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

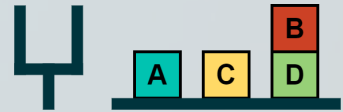
Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)

NONE



actions:

StaviNaBlok(C, A)

UzmiSaStola(C)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

# Zadatak 1 - Rešenje

state: **NaStolu(A)**, **NaStolu(C)**, NaStolu(D), **NaVrhu(C)**, NaVrhu(A),  
**RukaPrazna**, NaVrhu(B), Na(B, D)

goals:

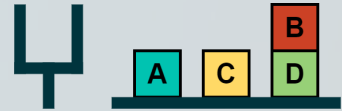
**Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna**

**Na(C, A)**

**NONE**

**URuci(C) & NaVrhu(A)**

**URuci(C)**



actions:

**StaviNaBlok(C, A)**

plan:

**SkiniSaBloka(B, A)**

**StaviNaBlok(B, D)**

**UzmiSaStola(C)**

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(A), NaVrhu(B), Na(B, D),  
**URuci(C)**

goals:

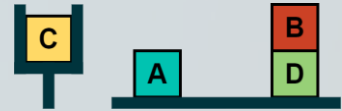
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

UzmiSaStola(C)

operator **UzmiSaStola(X)**:

preduslov: RukaPrazna & NaVrhu(X) & NaStolu(X)

ukloni: RukaPrazna & NaVrhu(X) & NaStolu(X)

dodaj: **URuci(X)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(A), NaVrhu(B), Na(B, D),  
URuci(C)

goals:

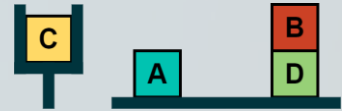
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

UzmiSaStola(C)



# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(A), NaVrhu(B), Na(B, D),  
URuci(C)

goals:

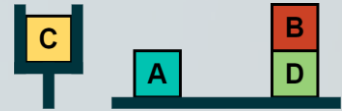
Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna

Na(C, A)

NONE

URuci(C) & NaVrhu(A)

URuci(C)



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

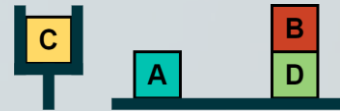
UzmiSaStola(C)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(A), NaVrhu(B), Na(B, D),  
URuci(C)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
NONE



actions:

StaviNaBlok(C, A)

plan:

SkiniSaBloka(B, A)

StaviNaBlok(B, D)

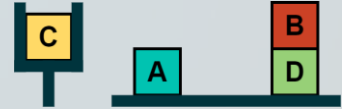
UzmiSaStola(C)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(A), NaVrhu(B), Na(B, D),  
URuci(C)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)  
**NONE**



actions:

**StaviNaBlok(C, A)**

plan:

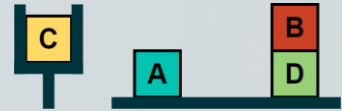
SkiniSaBloka(B, A)  
StaviNaBlok(B, D)  
UzmiSaStola(C)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), **NaVrhu(A)**, NaVrhu(B), Na(B, D),  
**URuci(C)**

goals:

**Na(C, A)** & **NaVrhu(C)** & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)



actions:

-

plan:

SkiniSaBlok(B, A)  
StaviNaBlok(B, D)  
UzmiSaStola(C)  
**StaviNaBlok(C, A)**

operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X)** & **NaVrhu(Y)**

ukloni: **URuci(X)** & **NaVrhu(Y)**

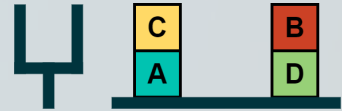
dodaj: **RukaPrazna** & **NaVrhu(X)** & **Na(X, Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(B), Na(B, D), **RukaPrazna**,  
**NaVrhu(C), Na(C, A)**

goals:

**Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna**  
**Na(C, A)**



actions:

-

plan:

**SkiniSaBloka(B, A)**  
**StaviNaBlok(B, D)**  
**UzmiSaStola(C)**  
**StaviNaBlok(C, A)**

operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

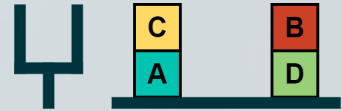
dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(B), Na(B, D), RukaPrazna,  
NaVrhu(C), Na(C, A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)



actions:

-

plan:

SkiniSaBlok(B, A)  
StaviNaBlok(B, D)  
UzmiSaStola(C)  
StaviNaBlok(C, A)

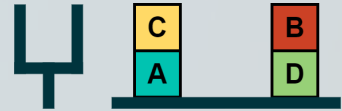


# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(B), Na(B, D), RukaPrazna,  
NaVrhu(C), Na(C, A)

goals:

Na(C, A) & NaVrhu(C) & NaStolu(A) & Na(B, D) & NaVrhu(B) & NaStolu(D) & RukaPrazna  
Na(C, A)



actions:

-

plan:

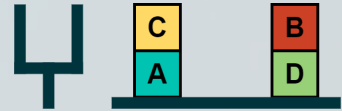
SkiniSaBloka(B, A)  
StaviNaBlok(B, D)  
UzmiSaStola(C)  
StaviNaBlok(C, A)

# Zadatak 1 - Rešenje

state: NaStolu(A), NaStolu(D), NaVrhu(B), Na(B, D), RukaPrazna,  
NaVrhu(C), Na(C, A)

goals:

-



actions:

-

plan:

SkiniSaBloka(B, A)  
StaviNaBlok(B, D)  
UzmiSaStola(C)  
StaviNaBlok(C, A)

## Zadatak 2 - Problem izbora



Uvodeći iste stavove za opis stanja i akcije za izmenu stanja iz zadatka 1, primenom STRIPS algoritma generisati niz akcija kojima bi se skup blokova u stanju prikazanom na levoj slici prebacio u stanje prikazano na desnoj slici. U hvataljci u jednom trenutku sme da se nađe maksimalno jedan blok. Na stolu može da se nađe proizvoljan broj blokova. Akcije se biraju po sledećem prioritetu: UzmiSaStola(X), SpustiNaSto(X), SkiniSaBloka(X,Y), StaviNaBlok(X,Y). Ciljno stanje se na stek dodaje po sledećem redosledu: NaStolu(B), Na(A, B), NaVrhu(A), RukaPrazna.



# Zadatak 2 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaVrhu}(B)$ ,  $\text{RukaPrazna}$

goals:

$\text{NaStolu}(B)$  &  $\text{Na}(A, B)$  &  $\text{NaVrhu}(A)$  &  $\text{RukaPrazna}$

$\text{NaStolu}(B)$

$\text{Na}(A, B)$

$\text{NaVrhu}(A)$

$\text{RukaPrazna}$



actions:

-

plan:

-

# Zadatak 2 - Rešenje

state: Na(B, A), NaStolu(A), NaVrhu(B), **RukaPrazna**



goals:

NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna

NaStolu(B)

Na(A, B)

NaVrhu(A)

**RukaPrazna**

actions:

-

plan:

-

# Zadatak 2 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaVrhu}(B)$ ,  $\text{RukaPrazna}$

goals:

$\text{NaStolu}(B)$  &  $\text{Na}(A, B)$  &  $\text{NaVrhu}(A)$  &  $\text{RukaPrazna}$

$\text{NaStolu}(B)$

$\text{Na}(A, B)$

$\text{NaVrhu}(A)$



actions:

-

plan:

-



# Zadatak 2 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaVrhu}(B)$ ,  $\text{RukaPrazna}$

goals:

$\text{NaStolu}(B)$  &  $\text{Na}(A, B)$  &  $\text{NaVrhu}(A)$  &  $\text{RukaPrazna}$

$\text{NaStolu}(B)$

$\text{Na}(A, B)$

$\text{NaVrhu}(A)$



actions:

-

plan:

-

# Zadatak 2 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

## NaVrhu(A)?



# Zadatak 2 - Rešenje

state:  $\text{Na}(\text{B}, \text{A}), \text{NaStolu}(\text{A}), \text{NaVrhu}(\text{B}), \text{RukaPrazna}$

goals:

$\text{NaStolu}(\text{B}) \ \& \ \text{Na}(\text{A}, \text{B}) \ \& \ \text{NaVrhu}(\text{A}) \ \& \ \text{RukaPrazna}$

$\text{NaStolu}(\text{B})$

$\text{Na}(\text{A}, \text{B})$

$\text{NaVrhu}(\text{A})$

**NONE**

**URuci(A)**



actions:

-

plan:

-

operator **SpustiNaSto(X):**

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**



actions:

**SpustiNaSto(A)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**



actions:  
**SpustiNaSto(A)**

plan:  
-

# Zadatak 2 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

**URuci(A)?**





# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**



goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**

**NaStolu(A)**

actions:

**SpustiNaSto(A)**

plan:

-

operator **UzmiSaStola(X):**

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**



goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**

**NaStolu(A)**

actions:

**SpustiNaSto(A)**

**UzmiSaStola(A)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A)**, **NaStolu(A)**, NaVrhu(B), RukaPrazna

goals:

**NaStolu(B)** & **Na(A, B)** & **NaVrhu(A)** & RukaPrazna

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna** & **NaVrhu(A)** & **NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**

**NaStolu(A)**



actions:

**SpustiNaSto(A)**

**UzmiSaStola(A)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**



actions:

**SpustiNaSto(A)**

**UzmiSaStola(A)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**



actions:

**SpustiNaSto(A)**

**UzmiSaStola(A)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & NaStolu(A)**

**RukaPrazna**

**NaVrhu(A)**



actions:

**SpustiNaSto(A)**

**UzmiSaStola(A)**

plan:

-



# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**



actions:  
**SpustiNaSto(A)**

plan:  
-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**



actions:  
**SpustiNaSto(A)**

plan:  
-

# Zadatak 2 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

**URuci(A)?**



# Zadatak 2 - Rešenje

state:  $\text{Na}(B, A)$ ,  $\text{NaStolu}(A)$ ,  $\text{NaVrhu}(B)$ ,  $\text{RukaPrazna}$

goals:

$\text{NaStolu}(B)$  &  $\text{Na}(A, B)$  &  $\text{NaVrhu}(A)$  &  $\text{RukaPrazna}$

$\text{NaStolu}(B)$

$\text{Na}(A, B)$

$\text{NaVrhu}(A)$

NONE

$\text{URuci}(A)$

NONE

$\text{RukaPrazna}$  &  $\text{NaVrhu}(A)$  &  $\text{Na}(A, B)$

$\text{RukaPrazna}$

$\text{NaVrhu}(A)$

$\text{Na}(A, B)$



actions:

$\text{SpustiNaSto}(A)$

plan:

-

operator  $\text{SkiniSaBloka}(X, Y)$ :

preduslov:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{Na}(X, Y)$

ukloni:  $\text{RukaPrazna}$  &  $\text{NaVrhu}(X)$  &  $\text{Na}(X, Y)$

dodaj:  $\text{URuci}(X)$ ,  $\text{NaVrhu}(Y)$

## Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**



goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & Na(A, B)**

**RukaPrazna**

**NaVrhu(A)**

**Na(A, B)**

actions:

**SpustiNaSto(A)**

**SkiniSaBloka(A, B)**

plan:

-

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**



goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & Na(A, B)**

**RukaPrazna**

**NaVrhu(A)**

**Na(A, B)**

actions:

**SpustiNaSto(A)**

**SkiniSaBloka(A, B)**

plan:

-

## Zadatak 2 - Rešenje

operator **UzmiSaStola(X)**:

preduslov: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

ukloni: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

dodaj: **URuci(X)**

operator **SpustiNaSto(X)**:

preduslov: **URuci(X)**

ukloni: **URuci(X)**

dodaj: **RukaPrazna & NaVrhu(X) & NaStolu(X)**

operator **SkiniSaBloka(X, Y)**:

preduslov: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

ukloni: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

dodaj: **URuci(X), NaVrhu(Y)**

operator **StaviNaBlok(X, Y)**:

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

Na(A, B)?





# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & Na(A, B)**

**RukaPrazna**

**NaVrhu(A)**

**Na(A, B)**

**NONE**

**URuci(A) & NaVrhu(B)**

**URuci(A)**

**NaVrhu(B)**



actions:

**SpustiNaSto(A)**

**SkiniSaBloka(A, B)**

plan:

-

operator **StaviNaBlok(X, Y):**

preduslov: **URuci(X) & NaVrhu(Y)**

ukloni: **URuci(X) & NaVrhu(Y)**

dodaj: **RukaPrazna & NaVrhu(X) & Na(X, Y)**

# Zadatak 2 - Rešenje

state: **Na(B, A), NaStolu(A), NaVrhu(B), RukaPrazna**

goals:

**NaStolu(B) & Na(A, B) & NaVrhu(A) & RukaPrazna**

**NaStolu(B)**

**Na(A, B)**

**NaVrhu(A)**

**NONE**

**URuci(A)**

**NONE**

**RukaPrazna & NaVrhu(A) & Na(A, B)**

**RukaPrazna**

**NaVrhu(A)**

**Na(A, B)**

**NONE**

**URuci(A) & NaVrhu(B)**

**URuci(A)**

**NaVrhu(B)**



actions:

**SpustiNaSto(A)**

**SkiniSaBloka(A, B)**

**StaviNaBlok(A, B)**

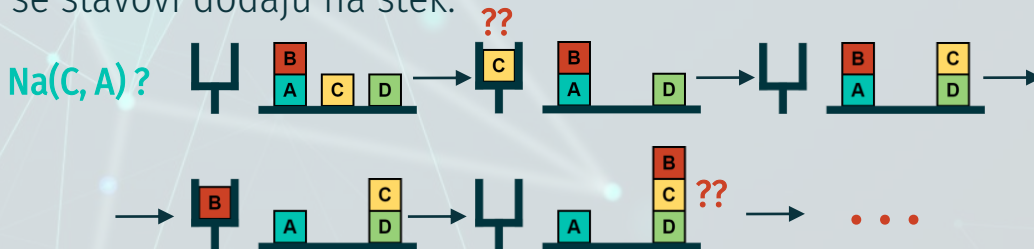
plan:

-

## Zadatak 2 - Rešenje

**Način izbora akcija** direktno može da utiče na rad algoritma. Potrebno je pažljivo odabrati heuristiku po kojoj se biraju akcije.

**Redosled dodavanja stavova složenog cilja na stek** direktno utiče na rad algoritma. Potrebno je pažljivo odabrati heuristiku po kojoj se stavovi dodaju na stek.



**Za samostalnu vežbu:** Izmeniti heuristiku za izbor akcija i redosled dodavanja stavova složenog cilja na stek u zadatku da bi se brže našlo rešenje.

# Zadatak za samostalnu vežbu - Kutije i figure



Dati su sledeći predikati za opis stanja blokova na stolu:

**UKutiji(N, X)** – označava da se u kutiji N nalazi figura X.

**Prazna(N)** – označava da se u kutiji N ne nalazi nijedna figura

**RukaPrazna** – Označava da nema figura u hvataljci

**URuci(X)** – Označava da se u hvataljci nalazi figura X

Date su sledeće akcije koje se mogu primeniti radi izmene stanja:

**UzmilzKutije(N, X)** – Hvataljkom se uzima figura X koji se nalazi u kutiji N.

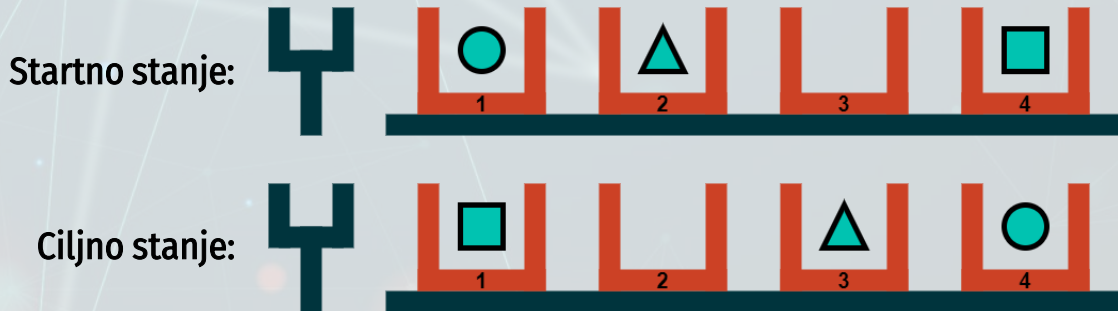
**SpustiUKutiju(N, X)** – Figura X koja se nalazi u hvataljci se spušta u kutiju N.



# Zadatak za samostalnu vežbu - Kutije i figure



Opisati startno i ciljno stanje prikazano na slici. Definirati potrebne liste stavova (Preduslov, Dodaj, Ukloni) za sve uvedene operatore. Primenom STRIPS algoritma generirati niz akcija kojima bi se skup figura prebacio iz startnog u ciljno stanje. U hvataljci u jednom trenutku sme da se nađe maksimalno jedna figura. U kutiji u jednom trenutku sme da se nađe maksimalno jedna figura.



# Zadatak za samostalnu vežbu - Hanojske kule



Dati su sledeći predikati za opis stanja diskova na stubovima:

**Na(N, D)** – označava da se na stubu N nalazi disk D.

Date su sledeće akcije koje se mogu primeniti radi izmene stanja:

**PomeriMaliDisk(N, M)** – Mali disk se sa stuba N pomera na stub M.

preduslov: Na(N, maliDisk)

ukloni: Na(N, maliDisk)

dodaj: Na(M, maliDisk)



**PomeriSrednjiDisk(N, M)** – Srednji disk se sa stuba N pomera na stub M.

preduslov: Na(N, srednjiDisk) & !Na(N, maliDisk) & !Na(M, maliDisk)

**PomeriVelikiDisk(N, M)** – Veliki disk se sa stuba N pomera na stub M.



# Zadatak za samostalnu vežbu - Hanojske kule



Opisati startno i ciljno stanje prikazano na slici. Dopuniti potrebne liste stavova (Preduslov, Dodaj, Ukloni) za sve uvedene operatore. Primenom STRIPS algoritma generisati niz akcija kojima bi se diskovi prebacili iz startnog u ciljno stanje. Samo jedan disk može da se pomera u datom trenutku i on mora da bude dostupan na stubu. Veći disk ne sme da se nađe iznad manjeg diska na stubu. Uvesti pretpostavku o zatvorenom svetu.





# PITANJA?

<http://ri4es.etf.rs/>

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.