



Електротехнички факултет, Универзитет у Београду

Катедра за рачунарску технику и информатику

Пројекат из предмета Експертски системи

професор: Бошко Николић
асистент: Дражен Драшковић

студент: Милан Бранковић
119/07

јун 2011.

Садржај

Опис проблема	3
Опис решења	5
Опис симулатора	6
Тестови	9

Опис проблема

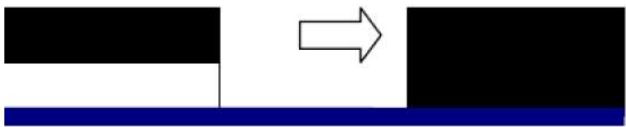
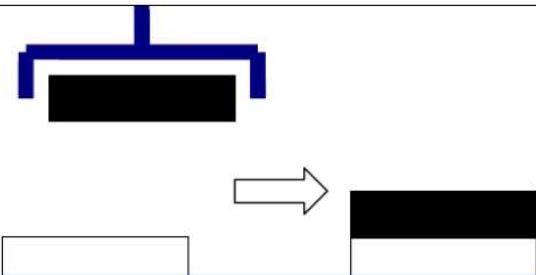
Замислите обојени свет блокова који се састоји од црно-белих блокова које може померати роботска рука. Блокови могу бити наслагани један на други, ненаслагани, подигнути и спуштени, као у класичном свету блокова. Поред тога они могу бити обојени, тачније доњи (нижи) блок може добити боју горњег (вишег) блока, ако су наслагани један на други.

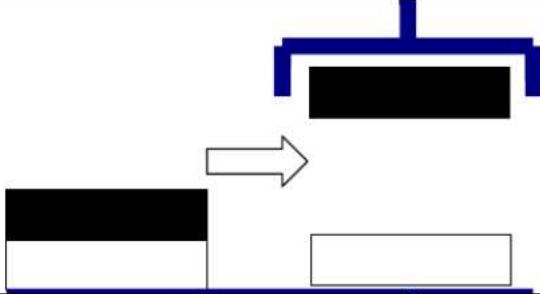
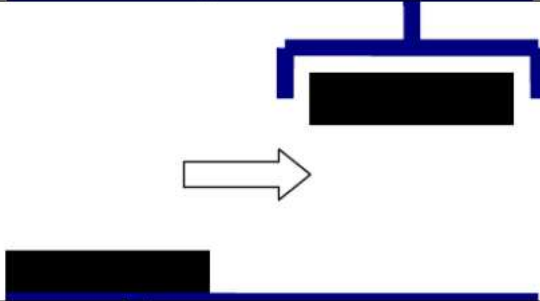
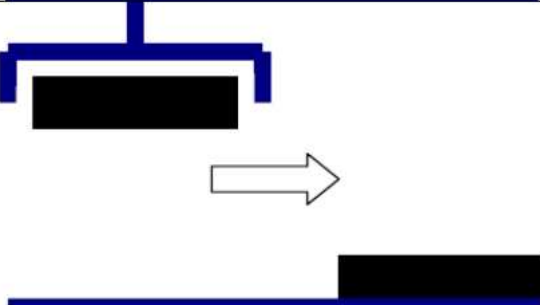
Ваш задатак је да коришћењем **СТРИПС алгоритма планирања**, имплементирате аутоматског играча у игри „Свет блокова“. Играч треба, у зависности од описа игре која му је дата скупом иницијалног стања света и циљног стања света (изражених као скуп предиката), да генерише поступак проналажења циљног стања почев од иницијалног стања. Поступак треба да садржи листу акција које се извршавају. Циљно стање може бити тако да **не постоје бели блокови** или да су **блокови поређани у неком задатом распореду**.

Листа предиката која описује стања у игри „Свет блокова“ је:

Предикат	Значење
on(x,y)	Блок x је на блоку y
onTable(x,y)	Блок x је на столу
clear(x)	Нема ничега на блоку x
holding(x)	Рука робота држи блок x
armEmpty	Рука робота је празна
black(x)	Блок x је црн
white(x)	Блок x је бео

Листа могућих акција:

	<i>assimilate (x, y)</i> Pre {Black(x), White(y), On (x,y)} Del {White (y)} Add {Black (y)}
	<i>stack (x, y)</i> Pre {Clear (y), Holding (x)} Del {Clear(y), Holding (x)} Add {ArmEmpty, On(x,y)}

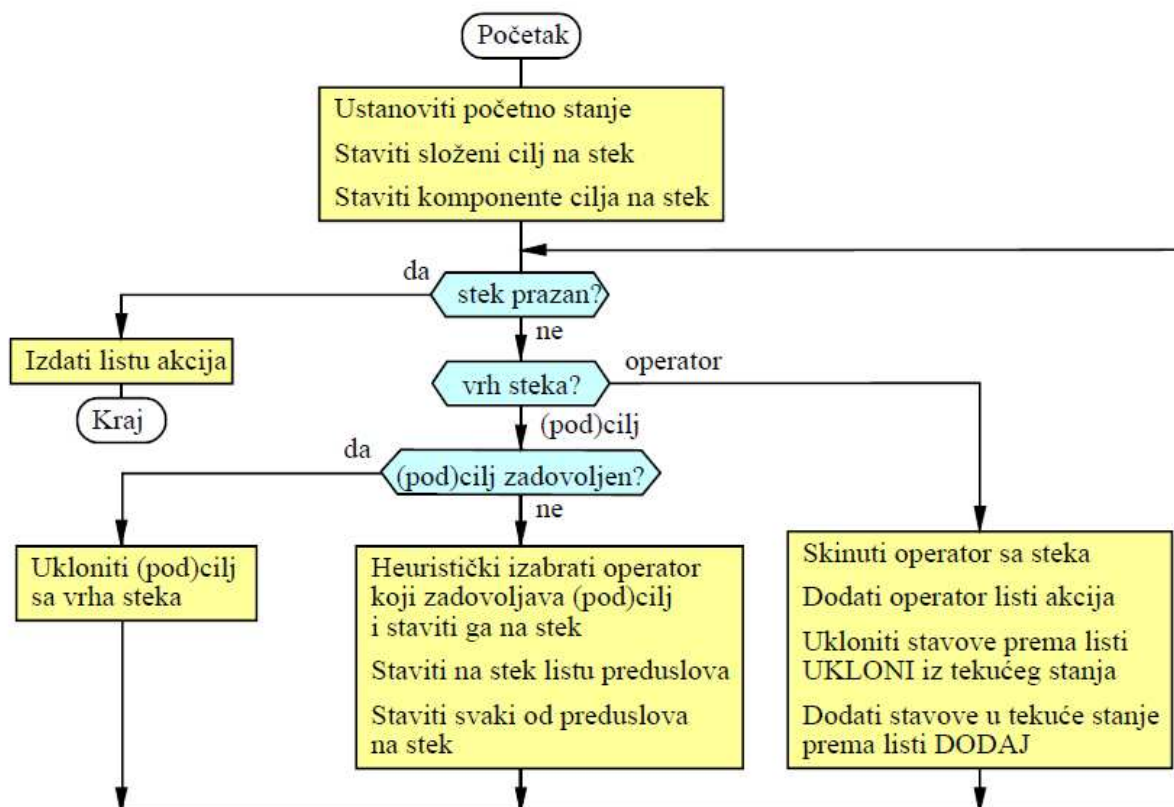
	<p><i>unstack (x, y)</i></p> <p>Pre { <i>On(x,y), Clear (x), ArmEmpty</i> }</p> <p>Del { <i>On(x,y), ArmEmpty</i> }</p> <p>Add { <i>Clear (y), Holding (x)</i> }</p>
	<p><i>pickup (x)</i></p> <p>Pre { <i>Clear (x), OnTable(x), ArmEmpty</i> }</p> <p>Del { <i>OnTable(x), ArmEmpty</i> }</p> <p>Add { <i>Holding (x)</i> }</p>
	<p><i>putdown (x)</i></p> <p>Pre { <i>Holding (x)</i> }</p> <p>Del { <i>Holding (x)</i> }</p> <p>Add { <i>OnTable(x), ArmEmpty</i> }</p>

Опис решења

Целокупан програм је написан у програмском језику JAVA. Програм има графички кориснички интерфејс (GUI) који омогућава кориснику лако тестирање. Интерфејс пружа кориснику следеће опције:

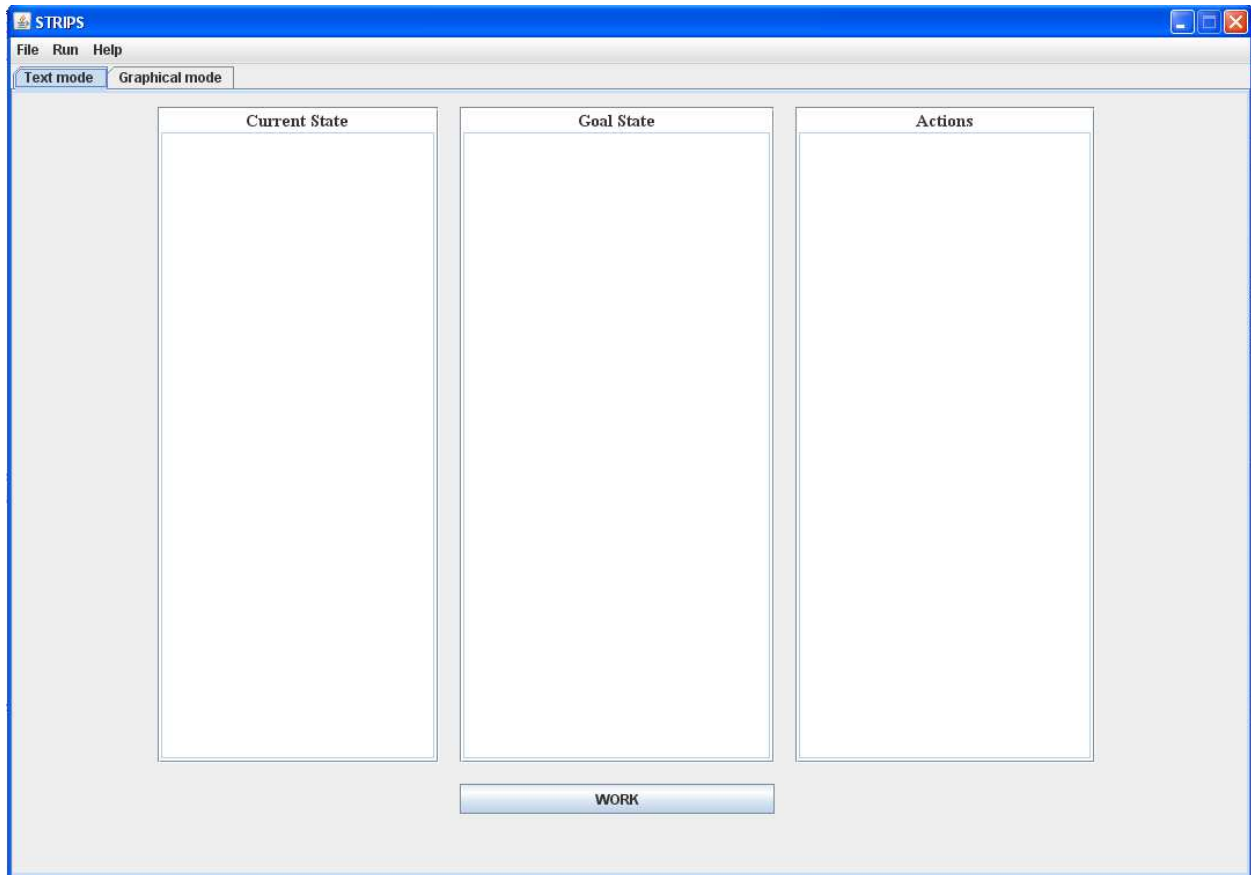
- опис иницијалног (почетног) и циљног стања и помоћи текстуалног описа (са предикатима), и помоћу графичког представљања (стварањем блокова одређене боје на одређеним позицијама на екрану)
- покретање алгоритма планирања што резултује исписивањем поступка (листа акција) у текстуалном формату
- извршење поступка који приказује графичку репрезентацију алгоритма, корак по корак, почевши од почетног све док се не стигне до циљног стања

Целокупно решење је имплементирано применом СТРИПС алгоритма планирања датог следећом сликом. При избору најбоље акције користи се хеуристика која бира најповољнију акцију која ће најбрже довести систем у циљно стање. Такође по могућству на почетку рада бира се акција ASSIMILATE која боји блокове.



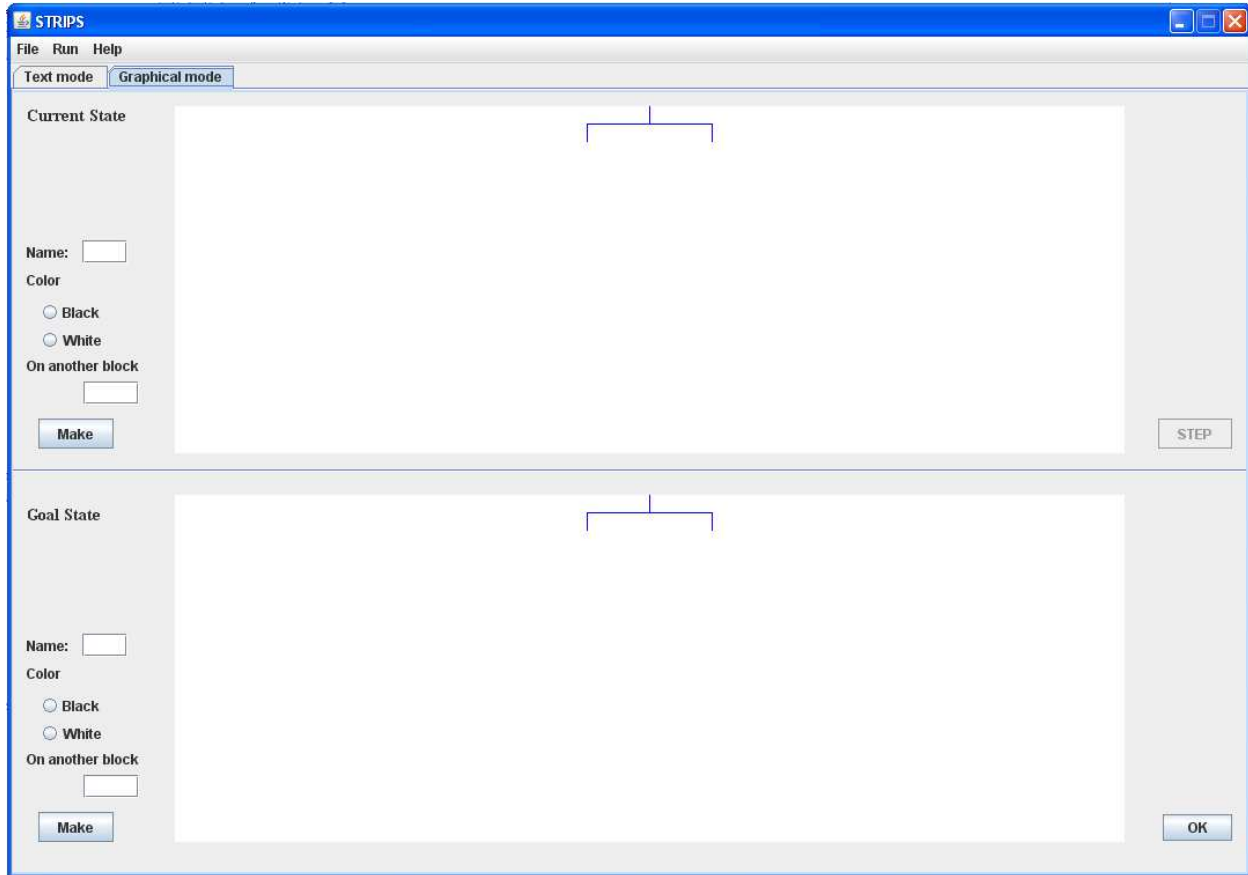
Опис симулатора

Изглед симулатора реализованог за текстуални режим



Корисник у овом моду може искористити предикате и задати тренутно (иницијално) и циљно стање. Предикати се пишу један испод другог, ради лакше прегледности. Најбоље је да се на почетку пишу предикати који се односе на боје, јер ће тако систем брже радити. По завршетку задавања стања кликом на дугме WORK, или избором ставке менија Run -> Work, у секцији Actions (акције) биће исписане акције које резултују превођењем блокова из тренутног у циљно стање.

Изглед симулатора у графичком режиму



Корисник у овом моду (режиму) има следећи избор:

- може да направи блок у текућем и у циљном стању тако што ће задати параметре тог блока: име (name), боју (black/white) и да ли се тај блок налази на неком другом блоку (on another block)
- након задавања почетног и циљног стања може да изврши задавање параметара за алгоритам кликом на дугме OK
- може да прати развој догађаја при преласку из почетног у циљно стање кликом на дугме STEP или избором ставке из падајућег менија Run -> Step. Дугме STEP, као и ставка падајућег менија постају активни након задавања параметара симулације, односно након клика на дугме OK

Поред ових режима кориснику су на располагању и следеће ставке падајућих менија:

- File -> New, избором ове ставке симулатор се враћа у почетно стање
- File -> Exit, избором ове ставке затвара се прозор у коме је отворен симулатор
- Help -> Help, избором ове ставке отвара се нови прозор који има излистане предикате и њихово значење

HelpDialog	
Predicates	Meaning
on(x,y)	Block x is on block y
onTable(x)	Block x is on table
clear(x)	There is nothing on block x
holding(x)	Robot arm is holding block x
armEmpty	Robot arm is empty
black(x)	Block x is black
white(x)	Block x is white

- Help -> About, избором ове ставке отвара се нови прозор који има аутора овог симулатора



Уколико корисник начини неку погрешну акцију, постоје упозорења да је нешто лоше одрађено, како за текстуални режим, тако и за графички.

Тестови

Коришћени су следећи тестови за тестирање:

1. Задатак 87 из збирке задатака из експертских система

CurrentState

on(B,A)
clear(B)
onTable(A)
clear(C)
onTable(C)
clear(D)
onTable(D)
armEmpty

GoalState

on(C,A)
clear(C)
onTable(A)
on(B,D)
clear(B)
onTable(D)
armEmpty

Actions

UNSTACK(B,A)
STACK(B,D)
PICKUP(C)
STACK(C,A)

2. Задатак 88 из збирке задатака из експертских система

CurrentState

onTable(A)
on(B,A)
clear(B)
armEmpty

GoalState

onTable(B)
on(A,B)
clear(A)
armEmpty

Actions

UNSTACK(B,A)
PUTDOWN(B)
PICKUP(A)
STACK(A,B)

3. Један блок на столу у почетном и крајњем стању

CurrentState

onTable(A)
clear(A)
armEmpty

GoalState

onTable(A)
clear(A)
armEmpty

Actions

4. Задатак из поставке пројекта

CurrentState

on(A,B)
 on(C,D)
 onTable(E)
 clear(A)
 clear(C)
 clear(E)
 onTable(B)
 onTable(D)
 onTable(E)
 black(A)
 black(B)
 white(C)
 white(D)
 white(E)
 armEmpty

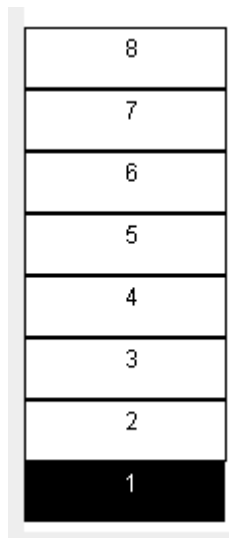
GoalState

black(A)
 black(B)
 black(C)
 black(D)
 black(E)
 onTable(A)
 onTable(B)
 onTable(C)
 onTable(D)
 onTable(E)
 clear(A)
 clear(B)
 clear(C)
 clear(D)
 clear(E)
 armEmpty

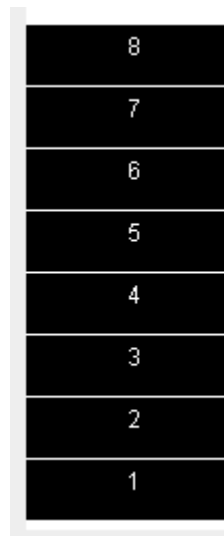
Actions

UNSTACK(C,D)
 PUTDOWN(C)
 UNSTACK(A,B)
 PUTDOWN(A)
 PICKUP(B)
 STACK(B,E)
 ASSIMILATE(B,E)
 PICKUP(A)
 STACK(A,D)
 ASSIMILATE(A,D)
 UNSTACK(A,D)
 STACK(A,C)
 ASSIMILATE(A,C)
 UNSTACK(B,E)
 PUTDOWN(B)
 UNSTACK(A,C)
 PUTDOWN(A)

5. Black block tower



CurrentState



GoalState

6. Black and white block tower

8
7
6
5
4
3
2
1

CurrentState

8
7
6
5
4
3
2
1

GoalState