

# Title

Stefan Åhman  
sahman@kth.se

Marcus Wallsttersson  
mwallst@kth.se

December 3, 2011

KTH Kista, Stockholm

## Innehållsförteckning

<b>1</b>	<b>Inledning</b>	<b>3</b>
<b>2</b>	<b>Problem och Syfte</b>	<b>3</b>
<b>3</b>	<b>Genomförande</b>	<b>3</b>
<b>4</b>	<b>Resultat</b>	<b>3</b>
	<b>Referenser</b>	<b>4</b>

## 1 Inledning

För att kunna kontrollera om en temporallogisk formel  $\phi$  gäller i ett visst tillstånd  $s$  i en given modell  $M$  kan man använda sig av en modellprovare. Detta programverktyg måste i denna laboration implementeras att hantera följande delmängd CTL-reglerna (Computation tree logic):

formler

För att kunna kontrollera om en temporallogisk formel  $\phi$  gäller i ett visst tillstånd  $s$  i en given modell  $M$  kan man använda sig av en modellprovare. Detta programverktyg måste i denna laboration implementeras att hantera följande delmängd CTL-reglerna (Computation tree logic):

## 2 Problem och Syfte

För att kunna kontrollera om en temporallogisk formel  $\phi$  gäller i ett visst tillstånd  $s$  i en given modell  $M$  kan man använda sig av en modellprovare. Detta programverktyg måste i denna laboration implementeras att hantera följande delmängd CTL-reglerna (Computation tree logic):

- fördjupa förståelsen för CTL och hur temporallogik kan användas för att specificera viktiga systemegenskaper.
- lära sig använda Prologs sökteknik för bevissökning.
- lära sig bygga enkla men nyttiga programverktyg som kan användas till systemverifikation.

## 3 Genomförande

## 4 Resultat

## Referenser