

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Reasoning on programs

The INFDEV Team @ HR

Hogeschool Rotterdam
Rotterdam, Netherlands

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Lecture topics

- We introduce conditional expressions
- We show how to verify properties on complex expressions

Conditional expressions

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Conditional expressions

- Sometimes we can make decisions within an expression
- The general form is `if CONDITION then S1 else S1' ; S` ($if_{S1CS1';S}$)
- If the condition is true, then we execute S1, otherwise S1'.
- After one of the possible executions we execute S.

$$\left\{ \begin{array}{ll} (PC, S) \xrightarrow{if_{S1CS1';S}} S1; S & \text{when } (PC, S) \xrightarrow{C} TRUE \\ (PC, S) \xrightarrow{if_{S1CS1';S}} S1'; S & \text{when } (PC, S) \xrightarrow{C} FALSE \end{array} \right.$$

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Conditional expressions

- "adult" if age ≥ 18 else "minor" = ?
- **Can you guess the results for age = 18 and age = 16?**

Conditional expressions

- `"adult" if age >= 18 else "minor" = ?`
- **Can you guess the results for `age = 18` and `age = 16`?**
- `age = 16: "minor"`
- `age = 18: "adult"`

Conditional expressions

- The ELSE block is not mandatory.
- Used when the execution of that part of code is optional depending on a condition.
- If the condition is TRUE we execute $S1$ and then S , otherwise only S .

$$\left\{ \begin{array}{ll} (PC, S) \xrightarrow{if S1 C; S} S1; S & \text{when } (PC, S) \xrightarrow{C} TRUE \\ (PC, S) \xrightarrow{if S1 C; S} S & \text{when } (PC, S) \xrightarrow{C} FALSE \end{array} \right.$$

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Reasoning on programs

- Sometimes we do not know exactly the values of all variables at all times
- The program may be too complex to allow it

Consider a throttle control system.

The throttle must never go under 1000RPM, or the engine stops and everybody dies.

The temperature must be kept under control, or the engine blows up and everybody dies.

```
throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

The question thus is: **could the code above cause everyone to die?**

throttle	temp
1000..10000	-20.0..400.0

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
1000..10000	-20.0..400.0

```
throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
?!?!?	-20.0..400.0

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Reasoning on programs

- We cannot list all possible combinations of variable values
- We cannot just “hope it works”

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Reasoning on programs

- We cannot list all possible combinations of variable values
- We cannot just “hope it works”
- We can reason in terms of conditions on variables

Reasoning on programs

- We partition the state based on the conditional
- $(temp > 350.0) \ \& \ (throttle > 2500)$ generates four states
 - $temp > 350$ and $throttle > 2500$
 - $temp \leq 350$ and $throttle > 2500$
 - $temp > 350$ and $throttle \leq 2500$
 - $temp \leq 350$ and $throttle \leq 2500$
- We study the semantics on each of these four states

`temp > 350 and throttle > 2500`

throttle	temp
<code>>2500.0 ..10000</code>	<code>>350.0 ..400.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

`temp > 350 and throttle > 2500`

throttle	temp
<code>>2500.0 ..10000</code>	<code>>350.0 ..400.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
<code>>1500.0..9000.0</code>	<code>>350.0..400.0</code>

`temp <= 350 and throttle > 2500`

throttle	temp
>2500.0..10000	-20.0..350.0

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```


`temp <= 350 and throttle > 2500`

throttle	temp
<code>>2500.0..10000</code>	<code>-20.0..350.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
<code>>2500.0..10000.0</code>	<code>-20.0..350.0</code>

`temp > 350 and throttle <= 2500`

throttle	temp
<code>1000..<code>>2500.0</code></code>	<code>>350..<code>400.0</code></code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

`temp > 350 and throttle <= 2500`

throttle	temp
<code>1000..>2500.0</code>	<code>>350..400.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
<code>1000..>2500.0</code>	<code>>350..400.0</code>

`temp <= 350 and throttle <= 2500`

throttle	temp
<code>1000..>2500.0</code>	<code>-20.0..>350.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
  throttle > 2500) else throttle
```

`temp <= 350 and throttle <= 2500`

throttle	temp
<code>1000..>2500.0</code>	<code>-20.0..350.0</code>

```
1 throttle = throttle - 1000 if (temp > 350.0) & (  
    throttle > 2500) else throttle
```

throttle	temp
<code>1000..>2500.0</code>	<code>-20.0..350.0</code>

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Reasoning on programs

- Each of the four states has a result
- We now merge the results

We now merge these states, knowing that each of them may actually happen:

throttle	temp
>1500.0..9000.0	>350.0..400.0
>2500.0..10000.0	-20.0..350.0
1000.0..>2500.0	>350..400.0
1000.0..>2500.0	-20.0..350.0

We now merge these states, knowing that each of them may actually happen:

throttle	temp
>1500.0..9000.0	>350.0..400.0
>2500.0..10000.0	-20.0..350.0
1000.0..>2500.0	>350..400.0
1000.0..>2500.0	-20.0..350.0

throttle	temp
1000.0..10000.0	-20.0..400.0

We know that the throttle will never go below 1500RPM, and we also know that if the temperature is above 350 degrees then maximum throttle is never above 9000RPM.

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

Nobody dies :)

Reasoning on
programs

The INFDEV
Team @ HR

Introduction

Conditional
expressions

Conditional
expressions

Reasoning on
programs

The best of luck, and thanks for the
attention!