



Semantics and Verification of Software

Winter Semester 2017/18

Lecture 14: Extension by Blocks and Procedures I (Operational Semantics)

Thomas Noll

Software Modeling and Verification Group

RWTH Aachen University

<http://moves.rwth-aachen.de/teaching/ws-1718/sv-sw/>

EvaSys	Vorlesung (dt./engl.)
RWTH Aachen WS17/18	apl. Prof. Dr.rer.nat. Thomas Noll Concurrency Theory (17ws-38073)



Markieren Sie so: Verwenden Sie bitte einen Kugelschreiber, rote Farbe unbedingt vermeiden!
 Mark: Please use a black ballpoint pen. Do not use red ink!
 Korrektur: Eintragungen außerhalb der markierten Felder fließen nicht in die Auswertung mit ein.
 Correction: Please observe the notes on the left when filling in the form in order to ensure optimal data collection.

Liebe Studierende,
 die RWTH Aachen hat das Ziel, Ihnen eine gute Lehre zu bieten.
 Dazu benötigen wir Ihre Mitarbeit und möchten Sie zu Ihrer
 Einschätzung zu der von Ihnen besuchten Veranstaltung befragen.
 Die Befragung und Auswertung erfüllen die datenschutzrechtlichen
 Bestimmungen; die Teilnahme ist anonym und freiwillig.
**Die Lehrenden sind verpflichtet, die Ergebnisse aus der
 Studentischen Lehrveranstaltungsbeurteilung mit den
 Studierenden in der Veranstaltung zu besprechen. Sollte dies
 nicht der Fall sein, wenden Sie sich bitte an:
 lehre@rwth-aachen.de.**
Dort wird Ihr Anliegen anonym behandelt.
 Legende:
 k.A. = keine Angabe

Dear Students,
 RWTH Aachen University aims to offer high standard of teaching. To
 this end, we depend on your cooperation and your assessment of
 the course attended. The survey and evaluation are carried out in
 accordance with the legal regulations for data protection;
 participation is anonymous and voluntary.
**Instructors are obliged to discuss the results of the student course
 evaluation with the students
 in the course. If this is not the case, please contact:
 lehre@rwth-aachen.de.**
Your concern will be handled anonymously.

Explanation:
 N/A = not applicable

1. Allgemein General Information	1.1 Geschlecht Gender	<input type="checkbox"/> weiblich female	<input type="checkbox"/> männlich male	<input type="checkbox"/> k.A. N/A				
	1.2 Nationalität Nationality	<input type="checkbox"/> deutsch (D) German (D)	<input type="checkbox"/> EU (ohne D) EU (excl. D)	<input type="checkbox"/> Non-EU				
	1.3 Derzeitiger Studiengang Course Degree	<input type="checkbox"/> Bachelor	<input type="checkbox"/> Master	<input type="checkbox"/> sonstiger other				
	1.4 Fachsemester Core Semester	<input type="checkbox"/> 1-2 7-8	<input type="checkbox"/> 3-4 >8	<input type="checkbox"/> 5-6				
	1.5 Wie viel Zeit verwenden Sie derzeit pro Woche für die Vor- und Nachbereitung dieser Veranstaltung? How much time do you currently spend on this course including preparation and follow up work?	<input type="checkbox"/> weniger als 1 Std. less than 1 hr.	<input type="checkbox"/> 1 bis 3 Std. 1 to 3 hrs.	<input type="checkbox"/> 3 bis 5 Std. 3 to 5 hrs.	<input type="checkbox"/> mehr als 9 Std. more than 9 hrs.			
	1.6 Die Veranstaltung interessiert mich. / I find the course interesting.	trifft zu strongly agree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	trifft nicht zu strongly disagree	<input type="checkbox"/>	k.A. N/A

2. Konzept der Vorlesung Lecture Concept	2.1 Die Lernziele der Vorlesung sind definiert. The learning goals of the lecture are defined.	trifft zu strongly agree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	trifft nicht zu strongly disagree	<input type="checkbox"/>	k.A. N/A	<input type="checkbox"/>
	2.2 Die Vorlesung hat eine klar erkennbare Struktur. The lecture is well structured.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2.3 Die zur Verfügung gestellten Materialien sind hilfreich. The materials provided are helpful.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2.4 Die ausgewählten Beispiele sind hilfreich. The examples chosen are helpful.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2.5 Es werden Zusammenfassungen an sinnvollen Stellen gemacht. Lecture material is summarized at appropriate intervals.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	2.6 Der Schwierigkeitsgrad ist ... The degree of difficulty is ...	<input type="checkbox"/> angemessen appropriate	<input type="checkbox"/> zu schwer too difficult	<input type="checkbox"/> zu leicht too easy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

EvaSys	Vorlesung (dt./engl.)
--------	-----------------------

[Fortsetzung]
[continued]

**3. Vermittlung und
Verhalten
Instruction and
Behavior**

2.7 Ich bewerte das Konzept der Vorlesung mit ... I would evaluate the lecture concept as ...	<input type="checkbox"/> 1 - sehr gut very good	<input type="checkbox"/> 2 - gut good	<input type="checkbox"/> 3 - befriedigend satisfactory	<input type="checkbox"/> 4 - ausreichend sufficient	<input type="checkbox"/> 5 - mangelhaft poor			
Die Dozentin/der Dozent ... The lecturer ...	trifft zu strongly agree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	trifft nicht zu strongly disagree	<input type="checkbox"/>	k.A. N/A	<input type="checkbox"/>
3.1 ... erklärt den Stoff verständlich. ... explains the subject matter clearly.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.2 ... geht auf Verständnisfragen ein. ... is willing to answer questions.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.3 ... berücksichtigt unterschiedliche Kenntnisstände der Studierenden. ... considers students' different levels of knowledge.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.4 ... schafft es, mich für den Vorlesungsstoff zu begeistern. / ... engages my interest in the topic.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.5 ... spricht angemessen laut und deutlich. ... speaks audibly and clearly.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.6 ... ist gut vorbereitet. / ... is well prepared.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.7 ... ist außerhalb der Vorlesung ansprechbar. ... is available outside of the lecture.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.8 ... setzt Medien ein, die zum Verständnis beitragen. ... uses media that contribute to students' understanding.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.9 Das Tempo ist ... The pace is ...	<input type="checkbox"/> angemessen appropriate	<input type="checkbox"/> zu hoch too fast	<input type="checkbox"/> zu niedrig too slow	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3.10 Ich gebe der Dozentin/dem Dozenten die Gesamtnote I would evaluate the lecturer as ...	<input type="checkbox"/> 1 - sehr gut very good	<input type="checkbox"/> 2 - gut good	<input type="checkbox"/> 3 - befriedigend satisfactory	<input type="checkbox"/> 4 - ausreichend sufficient	<input type="checkbox"/> 5 - mangelhaft poor			

**4. Rahmen-
bedingungen
General
Conditions**

4.1 Der zeitliche Rahmen der Vorlesung wird eingehalten. The lecture begins and ends on time.	trifft zu strongly agree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	trifft nicht zu strongly disagree	<input type="checkbox"/>	k.A. N/A	<input type="checkbox"/>
4.2 Die Anzahl der Sitzplätze ist ... The number of seats is ...	<input type="checkbox"/> angemessen appropriate	<input type="checkbox"/> zu groß too much	<input type="checkbox"/> zu klein too few	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4.3 Wie oft hat die Vorlesung an regulären Terminen gar nicht stattgefunden? (Vorlesungsfreie Tage sind nicht gemeint!) How often was the lecture cancelled on regularly scheduled dates? (Lecture-free days are not included!)	<input type="checkbox"/> 0 x	<input type="checkbox"/> 1 x	<input type="checkbox"/> 2 x	<input type="checkbox"/> 3 x	<input type="checkbox"/> 4 x	<input type="checkbox"/> >4 x	<input type="checkbox"/>	<input type="checkbox"/>
4.4 Wie häufig wurde die Vorlesung nicht von der angegebenen Person gehalten? How many times did you have a substitute teacher?	<input type="checkbox"/> 0 x	<input type="checkbox"/> 1 x	<input type="checkbox"/> 2 x	<input type="checkbox"/> 3 x	<input type="checkbox"/> 4 x	<input type="checkbox"/> >4 x	<input type="checkbox"/>	<input type="checkbox"/>
4.5 Falls sich die Dozentin/der Dozent vertreten lassen hat, war die Vertretung geeignet? In the event of a substitute teacher, was the substitute suitable?	<input type="checkbox"/> ja yes	<input type="checkbox"/> nein no	<input type="checkbox"/> k.A. N/A	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Extension by Blocks and Procedures

Outline of Lecture 14

Extension by Blocks and Procedures

Extending the Syntax

New Semantic Domains

Execution Relation

Extension by Blocks and Procedures

Blocks and Procedures

- Extension of WHILE by nested **blocks** with local **variables** and recursive **procedures**

Extension by Blocks and Procedures

Blocks and Procedures

- Extension of WHILE by nested **blocks** with local **variables** and recursive **procedures**
 - Simple memory model ($\Sigma := \{\sigma \mid \sigma : Var \rightarrow \mathbb{Z}\}$) not sufficient any more as variables can occur in several **instances**
- ⇒ Involves new semantic concepts:
- variable and procedure **environments**
 - **locations** (memory addresses) and **stores** (memory states)

Extension by Blocks and Procedures

Blocks and Procedures

- Extension of WHILE by nested **blocks** with local **variables** and recursive **procedures**
 - Simple memory model ($\Sigma := \{\sigma \mid \sigma : Var \rightarrow \mathbb{Z}\}$) not sufficient any more as variables can occur in several **instances**
- ⇒ Involves new semantic concepts:
- variable and procedure **environments**
 - **locations** (memory addresses) and **stores** (memory states)
- Important: **scope** of variable and procedure identifiers
- static scoping**: scope of identifier = **declaration environment**
(also: “lexical” scoping; here)
- dynamic scoping**: scope of identifier = **calling environment**
(old Algol/Lisp dialects)

Extension by Blocks and Procedures

Static and Dynamic Scoping

Example 14.1

```
begin
  var x; var y;
  proc P is y := x end;
  x := 1;
  begin
    var x;
    x := 2;
    call P
  end
end
```

Extension by Blocks and Procedures

Static and Dynamic Scoping

Example 14.1

```
begin
  var x; var y;
  proc P is y := x end;
  x := 1;
  begin
    var x;
    x := 2;
    call P
  end
end
```

static scoping $\Rightarrow y = 1$

Extension by Blocks and Procedures

Static and Dynamic Scoping

Example 14.1

```
begin
  var x; var y;
  proc P is y := x end;
  x := 1;
  begin
    var x;
    x := 2;
    call P
  end
end
```

static scoping $\Rightarrow y = 1$

dynamic scoping $\Rightarrow y = 2$

Extending the Syntax

Outline of Lecture 14

Extension by Blocks and Procedures

Extending the Syntax

New Semantic Domains

Execution Relation

Extending the Syntax

Extending the Syntax

Syntactic categories:

Category	Domain	Meta variable
Procedure identifiers	$PVar = \{P, Q, \dots\}$	P
Procedure declarations	$PDec$	p
Variable declarations	$VDec$	v
Commands (statements)	Cmd	c

Extending the Syntax

Extending the Syntax

Syntactic categories:

Category	Domain	Meta variable
Procedure identifiers	$PVar = \{P, Q, \dots\}$	P
Procedure declarations	$PDec$	p
Variable declarations	$VDec$	v
Commands (statements)	Cmd	c

Context-free grammar:

$p ::= \text{proc } P \text{ is } c \text{ end}; p \mid \varepsilon \in PDec$

$v ::= \text{var } x; v \mid \varepsilon \in VDec$

$c ::= \text{skip} \mid x := a \mid c_1; c_2 \mid \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end} \mid \text{while } b \text{ do } c \text{ end} \mid$
 $\text{call } P \mid \text{begin } v \ p \ c \ \text{end} \in Cmd$

Extending the Syntax

Extending the Syntax

Syntactic categories:

Category	Domain	Meta variable
Procedure identifiers	$PVar = \{P, Q, \dots\}$	P
Procedure declarations	$PDec$	p
Variable declarations	$VDec$	v
Commands (statements)	Cmd	c

Context-free grammar:

$p ::= \text{proc } P \text{ is } c \text{ end}; p \mid \varepsilon \in PDec$

$v ::= \text{var } x; v \mid \varepsilon \in VDec$

$c ::= \text{skip} \mid x := a \mid c_1; c_2 \mid \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end} \mid \text{while } b \text{ do } c \text{ end} \mid$
 $\text{call } P \mid \text{begin } v \ p \ c \ \text{end} \in Cmd$

- All used variable/procedure identifiers have to be declared
- Identifiers declared within a block must be distinct

New Semantic Domains

Outline of Lecture 14

Extension by Blocks and Procedures

Extending the Syntax

New Semantic Domains

Execution Relation

New Semantic Domains

Locations and Stores

- So far: **states** $\Sigma = \{\sigma \mid \sigma : \text{Var} \rightarrow \mathbb{Z}\}$

New Semantic Domains

Locations and Stores

- So far: **states** $\Sigma = \{\sigma \mid \sigma : Var \rightarrow \mathbb{Z}\}$
- Now: explicit control over all (nested) **instances** of a variable:
 - **variable environments**

$$VEnv := \{\rho \mid \rho : Var \dashrightarrow Loc\}$$

(partial function to maintain **declaredness** information)

- **locations**

$$Loc := \mathbb{N}$$

- **stores**

$$Sto := \{\sigma \mid \sigma : Loc \dashrightarrow \mathbb{Z}\}$$

(partial function to maintain **allocation** information)

New Semantic Domains

Locations and Stores

- So far: **states** $\Sigma = \{\sigma \mid \sigma : Var \rightarrow \mathbb{Z}\}$
- Now: explicit control over all (nested) **instances** of a variable:
 - **variable environments**

$$VEnv := \{\rho \mid \rho : Var \dashrightarrow Loc\}$$

(partial function to maintain **declaredness** information)

- **locations**

$$Loc := \mathbb{N}$$

- **stores**

$$Sto := \{\sigma \mid \sigma : Loc \dashrightarrow \mathbb{Z}\}$$

(partial function to maintain **allocation** information)

⇒ **Two-level access** to a variable $x \in Var$:

1. determine current memory location of x :

$$l := \rho(x)$$

2. reading/writing access to σ at location l

- Thus: previous **state** information represented as $\sigma \circ \rho$

Procedure Environments and Declarations

- **Effect of procedure call** determined by its body and variable and procedure environment of its declaration:

$$PEnv := \{ \pi \mid \pi : PVar \dashrightarrow Cmd \times VEnv \times PEnv \}$$

denotes the set of **procedure environments**

New Semantic Domains

Procedure Environments and Declarations

- **Effect of procedure call** determined by its body and variable and procedure environment of its declaration:

$$PEnv := \{\pi \mid \pi : PVar \dashrightarrow Cmd \times VEnv \times PEnv\}$$

denotes the set of **procedure environments**

- **Effect of declaration**: update of environment (and store)

$$\text{upd}_v[\cdot] : VDec \times VEnv \times Sto \rightarrow VEnv \times Sto$$

$$\text{upd}_v[\text{var } x ; v](\rho, \sigma) := \text{upd}_v[v](\rho[x \mapsto l_x], \sigma[l_x \mapsto 0])$$

$$\text{upd}_v[\varepsilon](\rho, \sigma) := (\rho, \sigma)$$

$$\text{upd}_p[\cdot] : PDec \times VEnv \times PEnv \rightarrow PEnv$$

$$\text{upd}_p[\text{proc } P \text{ is } c \text{ end}; \rho](\rho, \pi) := \text{upd}_p[\rho](\rho, \pi[P \mapsto (c, \rho, \pi)])$$

$$\text{upd}_p[\varepsilon](\rho, \pi) := \pi$$

where $l_x := \min\{l \in Loc \mid \sigma(l) = \perp\}$

Execution Relation

Outline of Lecture 14

Extension by Blocks and Procedures

Extending the Syntax

New Semantic Domains

Execution Relation

Execution Relation

Execution Relation I

Definition 14.2 (Execution relation)

For $c \in \mathit{Cmd}$, $\sigma, \sigma' \in \mathit{Sto}$, $\rho \in \mathit{VEnv}$, and $\pi \in \mathit{PEnv}$, the **execution relation** $(\rho, \pi) \vdash \langle c, \sigma \rangle \rightarrow \sigma'$ (“in environment (ρ, π) , statement c transforms store σ into σ' ”) is defined by the following rules:

$$\begin{array}{c} \text{(skip)} \frac{}{(\rho, \pi) \vdash \langle \text{skip}, \sigma \rangle \rightarrow \sigma} \\ \text{(asgn)} \frac{\langle a, \sigma \circ \rho \rangle \rightarrow z}{(\rho, \pi) \vdash \langle x := a, \sigma \rangle \rightarrow \sigma[\rho(x) \mapsto z]} \\ \text{(seq)} \frac{(\rho, \pi) \vdash \langle c_1, \sigma \rangle \rightarrow \sigma' \quad (\rho, \pi) \vdash \langle c_2, \sigma' \rangle \rightarrow \sigma''}{(\rho, \pi) \vdash \langle c_1 ; c_2, \sigma \rangle \rightarrow \sigma''} \\ \text{(if-t)} \frac{\langle b, \sigma \circ \rho \rangle \rightarrow \text{true} \quad (\rho, \pi) \vdash \langle c_1, \sigma \rangle \rightarrow \sigma'}{(\rho, \pi) \vdash \langle \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}, \sigma \rangle \rightarrow \sigma'} \end{array}$$

Execution Relation

Execution Relation II

Definition 14.2 (Execution relation; continued)

$$\frac{\langle b, \sigma \circ \rho \rangle \rightarrow \text{false} \quad (\rho, \pi) \vdash \langle c_2, \sigma \rangle \rightarrow \sigma'}{\text{(if-f)} \quad (\rho, \pi) \vdash \langle \text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}, \sigma \rangle \rightarrow \sigma'}$$

$$\frac{\langle b, \sigma \circ \rho \rangle \rightarrow \text{false}}{\text{(wh-f)} \quad (\rho, \pi) \vdash \langle \text{while } b \text{ do } c \text{ end}, \sigma \rangle \rightarrow \sigma}$$

$$\frac{\langle b, \sigma \circ \rho \rangle \rightarrow \text{true} \quad (\rho, \pi) \vdash \langle c, \sigma \rangle \rightarrow \sigma' \quad (\rho, \pi) \vdash \langle \text{while } b \text{ do } c \text{ end}, \sigma' \rangle \rightarrow \sigma''}{\text{(wh-t)} \quad (\rho, \pi) \vdash \langle \text{while } b \text{ do } c \text{ end}, \sigma \rangle \rightarrow \sigma''}$$

$$\frac{\text{(call)} \quad (\rho', \pi'[P \mapsto (c, \rho', \pi')]) \vdash \langle c, \sigma \rangle \rightarrow \sigma'}{(\rho, \pi) \vdash \langle \text{call } P, \sigma \rangle \rightarrow \sigma'} \quad \text{if } \pi(P) = (c, \rho', \pi')$$

$$\frac{\text{(block)} \quad \text{upd}_v[[v]](\rho, \sigma) = (\rho', \sigma') \quad \text{upd}_p[[p]](\rho', \pi) = \pi' \quad (\rho', \pi') \vdash \langle c, \sigma' \rangle \rightarrow \sigma''}{(\rho, \pi) \vdash \langle \text{begin } v \ p \ c \ \text{end}, \sigma \rangle \rightarrow \sigma''}$$

Execution Relation III

Remarks about rules (call) and (block):

- **Static scoping** is modelled in (call) by using the environments ρ' and π' (as determined in (block)) from the **declaration** site of procedure P (and not ρ and π from the **calling** site)
- In (call), the procedure environment associated with procedure P is extended by a P -entry to handle **recursive calls** of P :

$$\pi'[P \mapsto (c, \rho', \pi')]$$

Execution Relation

Execution Relation IV

Example 14.3

```
begin
  var x; var y;
  proc F is
    begin
      var z;
      z:=x;
      if z=1 then skip
        else x:=x-1; call F; y:=z*y end
    end
  end;
  x:=2; y:=1; call F
end
```

Diagram illustrating the execution relation for the code. Brackets on the right group the code into blocks: c_0 for the outermost block, c_1 for the inner block, c_F for the procedure body, and p for the call statement. A vertical bracket on the far right groups the entire code as c .

Let $\sigma_\emptyset(l) = \rho_\emptyset(x) = \pi_\emptyset(P) = \perp$ for all $l \in Loc, x \in Var, P \in PVar$

Notation: $\sigma_{ijkl} \Leftrightarrow \sigma(0) = i, \sigma(1) = j, \sigma(2) = k, \sigma(3) = l$

Derivation tree for $(\rho_\emptyset, \pi_\emptyset) \vdash \langle c, \sigma_\emptyset \rangle \rightarrow \sigma_{1221}$: on the board