

Vertalerboom 1 juli 2008

blad 1/17

(4 bladen in
dit deel)

Vraag 1. a

$$S \rightarrow AB \mid AC$$

$$A \rightarrow Aa \mid C \mid \epsilon$$

$$B \rightarrow Bb \mid C \mid \epsilon$$

$$C \rightarrow Cc \mid c$$

	first	follow
S	a, c, b	$\{\epsilon\}$ $\{\epsilon\} \cup \text{follow}(B) = \{\epsilon, b\}$
A	a, c	$\text{first}(B) \cup \text{follow}(S) \cup \text{first}(c) \cup \{\epsilon\}$ $\hookrightarrow \{a, b, c, \epsilon\}$
B	b, c	$\text{follow}(S) \cup \{b\}$ $\hookrightarrow \{\epsilon, b\}$
C	c	$\{c\} \cup \text{follow}(A) \cup \text{follow}(S)$ $\hookrightarrow \{a, b, c, \epsilon\}$

blad 2/17

b. LL(1)?

Neen, want links recursie ~~is~~ in

$A \rightarrow Aa,$

$B \rightarrow Bb$ en

$C \rightarrow Cc.$

Vraag 1b (cont'd) LR(0)?

①

$Z \rightarrow \cdot S \perp$	
$S \rightarrow \cdot AB$	
$S' \rightarrow \cdot AC$	
$A \rightarrow \cdot Aa$	
$A \rightarrow \cdot C$	
$A \rightarrow \cdot$	(reduce)
$C \rightarrow \cdot Cc$	⊗
$C \rightarrow \cdot c$	(shift)

Niet LR(0), want een shift/reduce ⊗

conflict. Maar misschien oplosbaar

in SLR(1)? Dan moet gelden:

$$c \notin \text{follow}(A) = \{a, b, c, \perp\}$$

en ~~z~~ zulks is niet het geval. Dus

ook niet SLR(1) ⊗

Vraag 1b (cont'd) LR(1)?

Z	→ · S	⊥	-
S	→ · AB	⊥	
S	→ · AC	⊥	
A	→ · A <u>a</u>	⊥, b, c, a	
A	→ · C	⊥, b, c, a	
A	→ ·	⊥, b, c, a	reduce
C	→ · Cc	⊥, b, c, a	
C	→ · c	⊥, b, c, a	shift

Shift/reduce conflict (wederom)

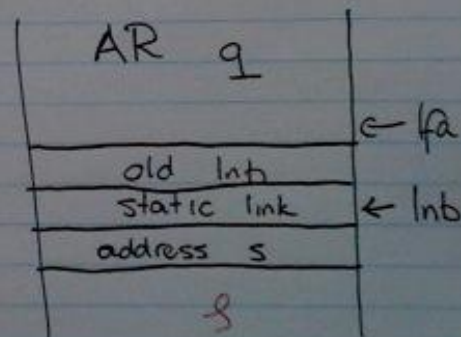
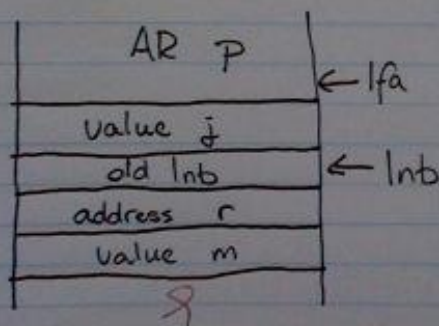
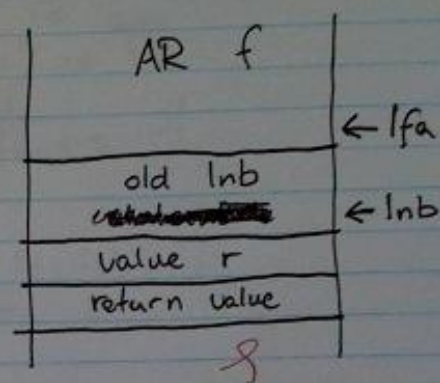
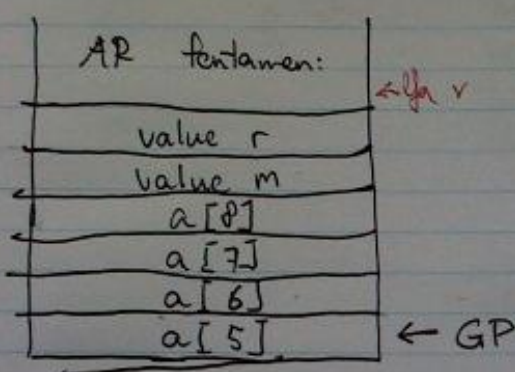
en $\{⊥, b, c\} \cap \{⊥, b, c\} \neq \emptyset$

dus niet LR(1). *g*

(5 bladen in dit deel)

blad 5/17

Vraag 2 a (Ik teken ze gewoon lekker allemaal, dat is straks handig)



blad 6/17

Vraag 2b

q.lab : $M[lfa] := env$
 $M[lfa+1] := lnb$
 $lnb := lfa$
 $lfa := lfa + 2$ ✗

(q.exit;) $lnb := M[lnb+1]$
 $lfa := lfa - 2$
 ✓ return

blad 7/17

Vraag 2c

① $f := a[r] + a[m]$

$R_0 := M[Inb - 1]$ // value r
 $R_1 := M[GP + 4]$ // value m

$R_2 := M[GP + R_0 - 5]$ // $a[r]$
 $R_3 := M[GP + R_1 - 5]$ // $a[m]$

$M[Inb - 2] := R_2 + R_3$ // return value

RETURN ^{komt bij} zit in exit_code // laatste statement in f.
✱

② $R_0 := M[M[Inb - 1]]$ // value s
 $R_1 := M[M[Inb] - 2]$ // value m

$M[ifa + 1] := R_1$ // zet r
 $ifa := ifa + 2$ // alloc r, returnval
CALL f.lab // f(m)
~~ifa := ifa + 1~~ // free r
 $R_2 := R_0 * M[ifa + 1]$ // s * f(m)
 $ifa := ifa - 2$ // free returnval

$R_3 := M[Inb] - 1$ // address r
 $M[R_3] := R_2$ // r := s + f(m)

RETURN // laatste statement

blad 8/17

Vraag 2c (cont'd)

③ $r := a[j] + m$

$R_0 := M[lnb + 1]$	// value j
$R_1 := M[GP + R_0 - 5]$	// value $a[j]$
$R_2 := M[lnb - 2]$	// value m
$R_3 := R_1 + R_2$	// $a[j] + m$
$M[M[lnb - 1]] := R_3$	// $r := a[j] + m$

! Let op: dit gaat hopeloos fout aangezien dit het eerste statement van P is, en de variabele j dus nog geen waarde heeft. *Heel slecht gezien, daar hadden dus nog ... voor moeten staan.*

(Of misschien de default-waarde, maar ik dacht niet dat Pascal dat deed.)

blad 9/17

↓ mag 2c (cont'd)

④ $q(r);$

$R_0 := \text{M}[lb-1]$ // ~~value~~ ^{addr} r

$M[ifa] := R_0$

$ifa := ifa + 1$

CALL $q.lab$

$ifa := ifa - 1$

RETURN

// alloc 5

// $q(r)$

// free 5

// laatste st. p

⑤ ~~q(b, r)~~

$M[ifa] := 6$

$M[ifa+1] := GP+5$

$ifa := ifa + 2$

CALL $p.lab$

$ifa := ifa - 2$

// ~~set~~ value 6

// addr r

// alloc

// $p(b, r)$

// free

blad 10/17

Vraag 3 (a) Default producties

(op bladen
in dit deel)

$Z \rightarrow S e o f s$

$S' \rightarrow$ ifs ~~E then S_i~~ E_p fis
 S_i terminant code

$E \rightarrow T X E$

$T \rightarrow$ ident

$X E \rightarrow$

$E_p \rightarrow$

$S_i \rightarrow$ assignment

$S_l \rightarrow S S_t$

$S_t \rightarrow$

Vraag 3 (b) follow

$$\begin{aligned} \text{follow}(Xt) &= \text{follow}(E) \quad \text{" + follow}(Xt) \text{"} \\ &= \{\text{thens, rpar}\} \quad \text{maar dat gaat} \\ &\quad \text{nergens over} \\ &\quad \text{naar links} \end{aligned}$$

$$\begin{aligned} \text{follow}(St) &= \text{follow}(SL) \\ &= \{\text{ends}\} \end{aligned}$$

(c) richters

$$\begin{aligned} \text{richters}(Xt \Rightarrow \text{op } TXt) &= \text{first}(\text{op } TXt) \\ &= \{\text{op}\} \end{aligned}$$

$$\begin{aligned} \text{richters}(Xt \Rightarrow \epsilon) &= \text{first}(\epsilon) + \text{follow}(Xt) \\ &= \{\} + \{\text{thens, rpar}\} \end{aligned}$$

$\{\text{op}\}$ en $\{\text{thens, rpar}\}$ zijn disjunct

blad 12/17

Vraag 3 (c) (cont'd) richters

$$\text{richters}(S'f \rightarrow sc \ S \ S't) = \text{first}(sc \ S \ S't) \\ = \{sc\}$$

$$\text{richters}(S'f \rightarrow \epsilon) = \text{first}(\epsilon) + \text{follow}(S't) \\ = \{\text{ends}\}$$

$\{sc\}$ en $\{\text{ends}\}$ zijn disjunct

blad 13/17

Vraag 3(d) implementatie

```
procedure PSI (keys : symbol set)
begin
    if sym ∈ {begs}
    then begin
        matchsym = match (begs, keys + {assignment, begs, ifs});
        PSI (keys + {ends});
        match (ends, keys);
    end
    else begin
        match (assignment, keys);
    end
end
end
```

blad 19/17

Vraag 3 (d) (cont'd)

```
procedure p, S1 ( keys : t symbolset )  
begin  
  p, S1' ( keys + { sc , ends } );  
  p, S1t ( keys );  
end
```

blad 15/19

Vraag 3 (d) (cont'd)

```
procedure p, S t (keys : tsymbolset)
begin
```

```
  if sym ∈ (sc)
  then begin
```

```
    match(sc, keys + (assignment, begs,
                     ifs));
```

```
    p, S t (keys + (sc, ends));
```

```
    p, S t (keys);
```

```
  end
```

```
  else begin
```

```
    (* skip *)
```

```
  end
```

```
end
```

Blad 16/17

Vraag 3 (d) (cont'd)

```
procedure match(s: tsymbol, keys: tsymbolset)
begin delete (key v ts); ✓
      if s == sym
      then begin
            next sym();
      end
      else begin
            delete insertion(s) (keys); ✗
      end
end
end
```


Vraag 3 (d) (cont'bl)

blad 17/17

begin (* hoofdprogramma *)

initScanner();

nextSym();

~~delete~~

PZ({L});

delete({L});

end (* main *)