

Imperative Programmierung Aufgabe A01

Erik Esins

Matrikelnummer: 219204576

Richard Sönke Högemann

Matrikelnummer:219204837

Sebastian Seifert

Matrikelnummer:219203783

Darleen Koblitz

Matrikelnummer:218203452

Nr1)

a)

```
1100000  s1 → s2  1 → 0 R
^
0100000  s2 → s2  1 → 1 R
^
0100000  s2 → s3  0 → 0 R
^
0100000  s3 → s4  0 → 1 L
^
0101000  s4 → s5  0 → 0 L
^
0101000  s5 → s5  1 → 1 L
^
0101000  s5 → s1  0 → 1 R
^
1101000  s1 → s2  1 → 0 R
^
1001000  s2 → s3  0 → 0 R
^
1001000  s3 → s3  1 → 1 R
^
1001000  s3 → s4  0 → 1 L
^
1001100  s4 → s4  1 → 1 L
^
1001100  s4 → s5  0 → 0 L
^
1001100  s5 → s1  0 → 1 R
^
1101100  s1 → s6  0 → 0 0
^
→ s6
```

b) 11111110

alter Zustand:

gelesenes Symbol:

geschriebenes Symbol:

Neuer Zustand:

s1	→ s1	→ s1	→ s1	→ s1	→ s1	→ s1	→ s1
1	→ 1	→ 1	→ 1	→ 1	→ 1	→ 1	→ 0
1	→ 1	→ 1	→ 1	→ 1	→ 1	→ 1	→ 0
s2	→ s3	→ s4	→ s5	→ s6	→ s7	→ s7	→ s8

Kopfrichtung: $R \rightarrow R \rightarrow R \rightarrow R \rightarrow R \rightarrow R \rightarrow 0$

Unsere Turinmaschine addiert die jeweils gelesenen Zustände und speichert sie in Form des Zustandes. Dadurch können wir sagen wie viele Einsen er addiert hat und sobald er keine weiteren Einsen mehr findet beendet er die addition.

Nr2)

S0 = Zuweisung			
Variable := Ausdruck	[Regel 1]		
„a“ := Ausdruck	[Regel 5]		
a := Ausdruck + Ausdruck	[Regel 4]		
a := Variable + Ausdruck	[Regel 2]		a := Ausdruck + Ausdruck + Ausdruck [Regel 4]
a := a + Ausdruck	[Regel 5]		a := Variable + Ausdruck + Ausdruck [Regel 2]
a := a + Ausdruck + Ausdruck	[Regel 4]		a := a + Ausdruck + Ausdruck [Regel 5]
	a := a + Variable + Ausdruck	[Regel 2]	
	a := a + b + Ausdruck	[Regel 6]	
	a := a + b + Konstante	[Regel 3]	
	a := a + b + 1	[Regel 7]	

Nr3)

Nr4)

Zuweisung:: Variable + „:=“ + Ausdruck
Ausdruck:: Ausdruck + „+“ + Ausdruck | Variable | Konstante
Variable:: „a“ | „b“
Konstante:: „1“ | „0“

Zuweisung
→ Variable → „:=“ → Ausdruck

Ausdruck
→ Ausdruck → + → Ausdruck
| → Variable
| → Konstante

Variable
→ „a“ →
| → „b“

Konstante
→ „1“ →
| → „0“ →

Nr5)

Adresse :: Name +“;“+Strasse+“;“+Stadt+“;“+[Telefonnummer|Faxnummer|Email-adresse]

Name:: Name einer Person|Name einer Firma

Strasse:: Strassenname + Hausnummer|Postfachnummer

Stadt:: Stadtname + Postleitzahl

→ Adresse →

→ Name → „;“ → Strasse → „;“ → Stadt → „;“ →
|→ Telefonnummer
|→ Faxnummer
|→ Email-adresse

Name

→ Name einer Person

|→ Name einer Firma

Strasse

Strassenname → Hausnummer

|→ Postfachnummer

Stadt

→ Stadtname → Postleitzahl

Nr 6)

a) Es handelt sich nicht um einen Algorithmus, da er nicht terminiert ist.

b)

Potenz
eingabe P N int erg = 1;
while (N>0) erg = erg * P; n = n - 1;
return erg

Fakultät
eingabe N int erg = 1;
while (N>0) erg = erg * n; n = n - 1;
return erg;

Sinus
eingabe X K int N = 0 float sum = 0;
while (N<K) sum = sum + (Potenz(-1,N))*((Potenz(X,(2N+1)))/(Fakultät(2N+1)));
return sum