Homework 5 Solutions

March 27, 2013

1 Attribute Grammar

a)

Production	Attribute Rules
Number -> 0	Number.value = 0
Number -> HighestBit List DoubleZero	HighestBit.position = List.Position + 1 Number.Value $= 2^{HighestBit.Position} + List.Value$
List0 -> 1 List1	List0.Position = 1 + List1.Position List0.Value $= 2^{List0.Position} + List1.Value$
List0 -> 0 List1	List1.Position = 1 + List1.Position List1.Value = List1.Value
List -> ε	List.Value = 0 List.Position = 1

b)

```
                        ┌─────────────┐
                        │   Number    │
                        │ {Value= 44} │
                        └─────────────┘
              ┌───────────────┼──────────────────┐
      ┌──────────────┐  ┌──────────────────────┐  ┌────────────┐
      │  HighestBit  │  │        List          │  │ DoubleZero │
      │{Position = 5}│  │{Position = 4; Value = 12}│ └────────────┘
      └──────────────┘  └──────────────────────┘
             │        ┌──────┴─────────┐              │
          ┌─────┐  ┌─────┐  ┌──────────────────────┐  ┌────┐
          │  1  │  │  0  │  │        List          │  │ 00 │
          └─────┘  └─────┘  │{Position = 3; Value = 12}│ └────┘
                            └──────────────────────┘
                              ┌────┴──────────┐
                           ┌─────┐  ┌──────────────────────┐
                           │  1  │  │        List          │
                           └─────┘  │{Position = 2; Value = 4}│
                                    └──────────────────────┘
                                      ┌────┴──────────┐
                                   ┌─────┐  ┌──────────────────────┐
                                   │  1  │  │        List          │
                                   └─────┘  │{Position = 1; Value = 0}│
                                            └──────────────────────┘
                                                      │
                                                  ┌───────┐
                                                  │ empty │
                                                  └───────┘
```
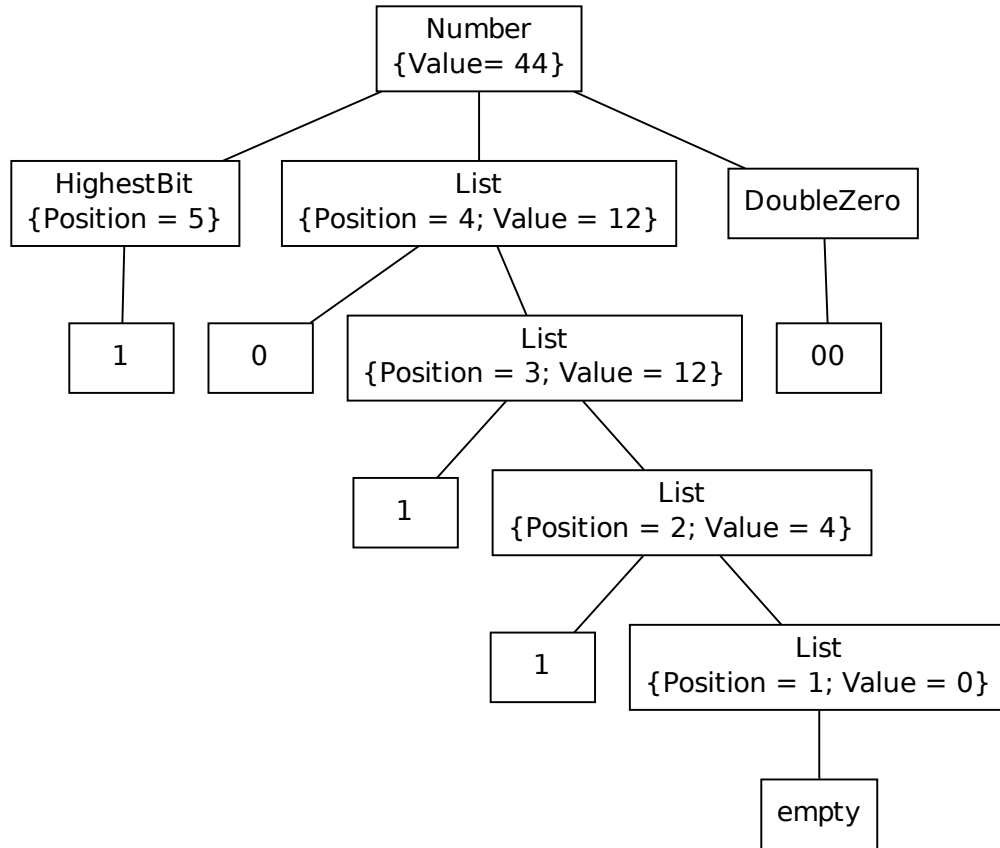
c) All but HighestBit.Position are synthesized attributes.
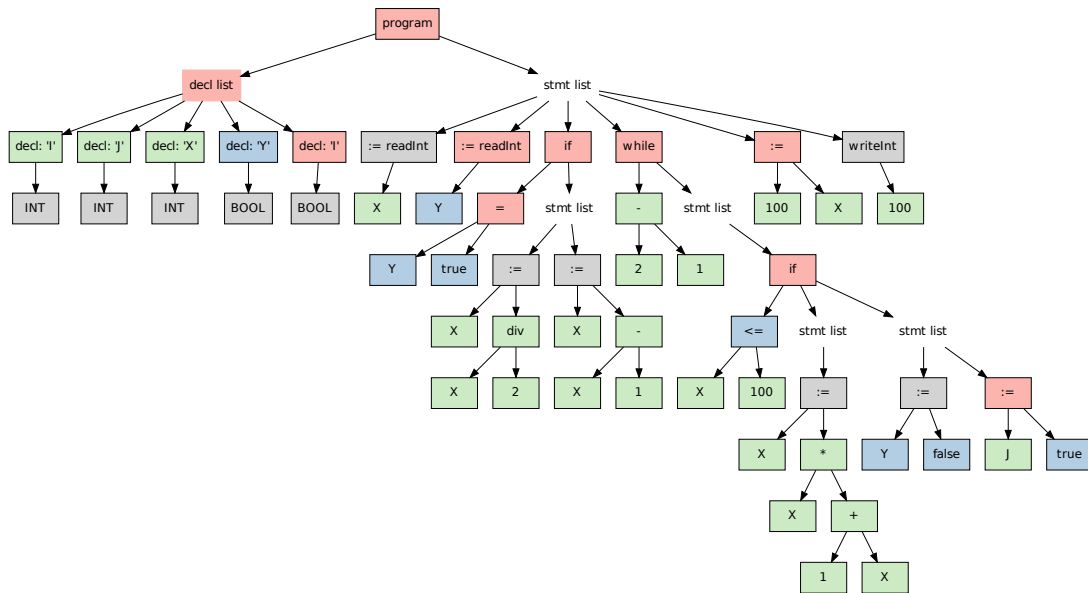
2   Tree Walk

```
load  a  -> r1
loab  b  -> r2
load  c  -> r3
mul  r1 ,  r3  -> r4
load  d  -> r5
load  4  -> r6
mul  r4 ,  r6  -> r7
mul  r2 ,  r2  -> r9
sub  r8 ,  r7  -> r9
i2i  r9  -> r5
```

3 AST

program
decl list — stmt list
decl: 'I' — decl: 'J' — decl: 'X' — decl: 'Y' — decl: 'I'
:= readInt — := readInt — if — while — := — writeInt
INT — INT — INT — BOOL — BOOL — X — Y — = — stmt list — - — stmt list — 100 — X — 100
Y — true — := — := — 2 — 1 — if
X — div — X — - — <= — stmt list — stmt list
X — 2 — X — 1 — X — 100 — := — := — :=
X — * — Y — false — J — true
X — +
1 — X

b)

I is declared twice.

Y is bool, and readInt cannot take bool on LHS.

"=" cannot have bool operands (no operators can have bool operands).

While condition expression must be type bool.

Boolean literals cannot be assigned to int variables.

Int literals cannot be assigned a value.

4 ILOC

```
        loadi    0 -> r0
        loadi    90 -> r1
loop:
        cmpLT    r0, r1 -> r2
        cbr      r2 -> body, exit
body:
        loadi    2 -> r3
        div      r0, r3 -> r4
        mul      r4, r3 -> r5
        cmpE     r5, r0 -> r6
        cbr      r6 -> then, loop
then:
        loadi    1, r7
        add      r0, r7 -> r0
        sub      r1, r7 -> r1
        jmp      loop
```