Orion Kostival
CSCI 400 A
2/28/2013
Homework #5

Problem 1

Original Grammar:

<program>	→	main { <stmts> }
<stmts>	→	<stmt>;
	→	<stmt>; <stmts>
<stmt>	→	<var> = <expr>
<expr>	→	<term> + <term>
	→	<term> - <term>
	→	<term>
<term>	→	<var> | const
<var>	→	a | b | c | d

a) **Augmented Grammar:**

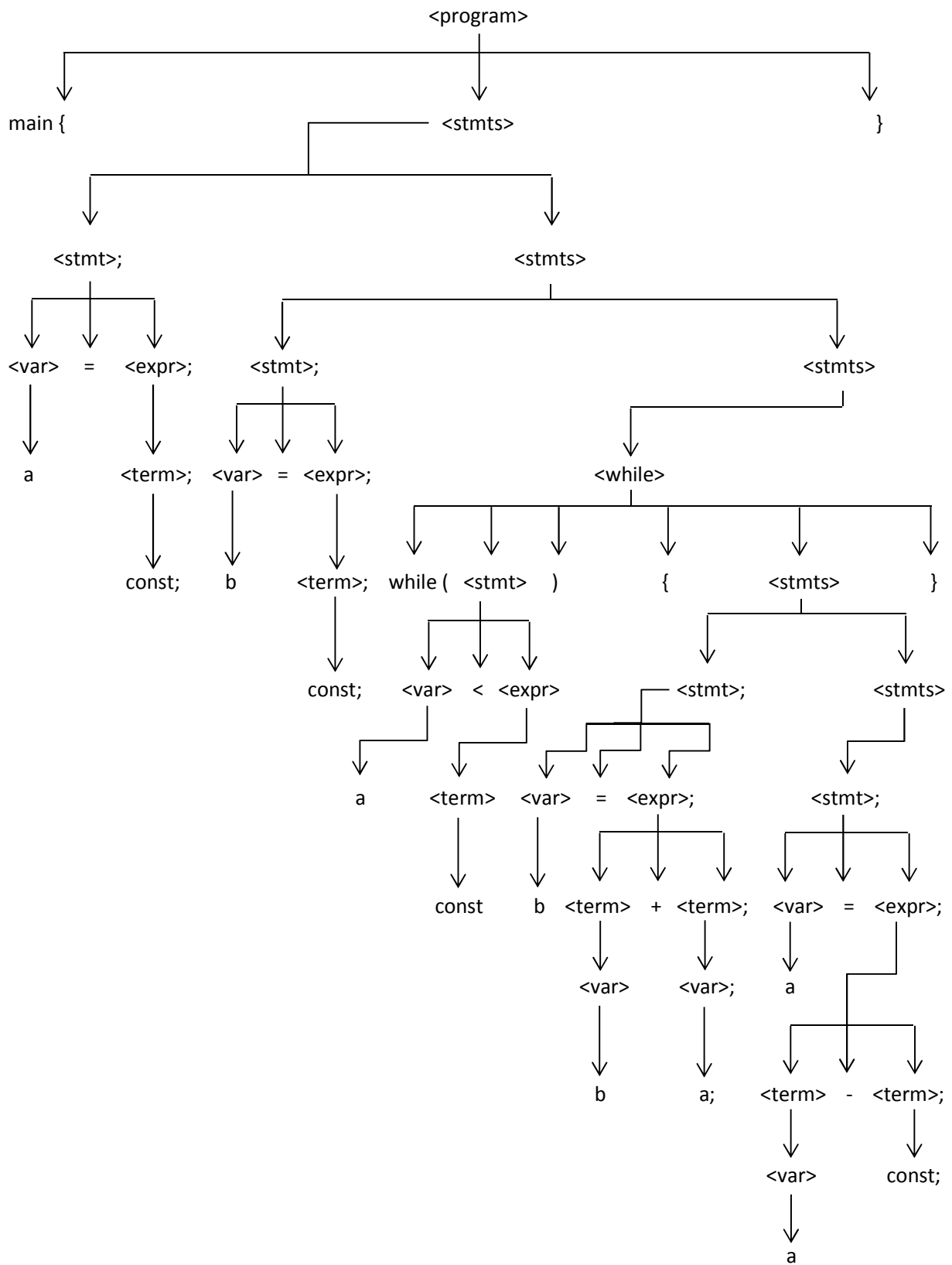|		
<while>	→	while (<stmt>) { <stmts> }
<for>	→	for (<stmt>; <stmt>; <stmt>) { <stmts> }
<stmts>	→	<for>
	→	<for> <stmts>
	→	<while>
	→	<while> <stmts>
<stmt>	→	<var> < <expr>

b) **Program 1:**

```
main
{
        a = const;
        b = const;
        while (a < const)
        {
                b = b + a;
                a = a – const;
        }
}
```

Leftmost Derivation:

<program> → main { <stmts> }

→ main { <stmt>; <stmts> }

→ main { <var> = <expr>; <stmts> }

→ main { a = <expr>; <stmts> }

→ main { a = <term>; <stmts> }

→ main { a = const; <stmts> }

→ main { a = const; <stmt>; <stmts> }

→ main { a = const; <var> = <expr>; <stmts> }

→ main { a = const; b = <expr>; <stmts> }

→ main { a = const; b = <term>; <stmts> }

→ main { a = const; b = const; <stmts> }

→ main { a = const; b = const; <while> }

→ main { a = const; b = const; while (<stmt>) { <stmts> }}

→ main { a = const; b = const; while (<var> < <expr>) { <stmts> }}

→ main { a = const; b = const; while (a < <expr>) { <stmts> }}

→ main { a = const; b = const; while (a < <term>) { <stmts> }}

→ main { a = const; b = const; while (a < const) { <stmts> }}

→ main { a = const; b = const; while (a < const) { <stmt>; <stmts> }}

→ main { a = const; b = const; while (a < const) { <var> = <expr>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = <expr>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = <term> + <term>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = <var> + <term>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = b + <term>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = b + <var>; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = b + a; <stmts> }}

→ main { a = const; b = const; while (a < const) { b = b + a; <stmt>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; <var> = <expr>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; a = <expr>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; a = <term> + <term>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; a = <term> - <term>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; a = <var> - <term>; }}

→ main { a = const; b = const; while (a < const) { b = b + a; a = a - <term>; }}

→ **main**
  {
    a = const;
    b = const;
    while (a < const)
    {
      b = b + a;
      a = a - const;
    }
  }

Parse Tree:

Program 2:
```
main
{
        a = const;
        b = const;
        for (a = const; a < const; a = a - const)
        {
                b = b + a;
        }
}
```

Leftmost Derivation:
```
<program> → main { <stmts> }
          → main { <stmt>; <stmts> }
          → main { <var> = <expr>; <stmts> }
          → main { a = <expr>; <stmts>  }
          → main { a = <term>; <stmts> }
          → main { a = const; <stmts> }
          → main { a = const; <stmt>; <stmts> }
          → main { a = const; <var> = <expr>; <stmts> }
          → main { a = const; b = <expr>; <stmts> }
          → main { a = const; b = <term>; <stmts> }
          → main { a = const; b = const; <stmts> }
          → main { a = const; b = const; <for> }
          → main { a = const; b = const; for (<stmt>; <stmt>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (<var> = <expr>; <stmt>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = <expr>; <stmt>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = <term>; <stmt>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; <stmt>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; <var> < <expr>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; a < <expr>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; a < <term>; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; a < const; <stmt>) { <stmts> } }
          → main { a = const; b = const; for (a = const; a < const; <var> = <expr>) { <stmts> } }
          → main { a = const; b = const; for (a = const; a < const; a = <expr>) { <stmts> } }
          → main {
                    a = const; b = const;
                    for (a = const; a < const;  a = <term> - <term>) { <stmts> }
                }
          → main {
                    a = const; b = const;
                    for (a = const; a < const;  a = <var> - <term>) { <stmts> }
                }
          → main {
                    a = const; b = const;
                    for (a = const; a < const;  a = a - <term>) { <stmts> }
                }
```

→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { <stmts> }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { <stmt>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { <var> = <expr>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { b = <expr>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { b = <term> + <term>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { b = <var> + <term>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { b = b + <term>; }
  }
→ main {
        a = const; b = const;
        for (a = const; a < const; a = a - const) { b = b + <var>; }
  }
→ **main**
  {
        a = const;
        b = const;
        for (a = const; a < const; a = a - const)
        {
             b = b + a;
        }
    }

Parse Tree:

<program>

main {   <stmts>   }

<stmt>;   <stmts>

<var> = <expr>;   <stmt>;   <stmts>

a   <term>;   <var> = <expr>;   <for>

const;   b   <term>;   for (<stmt>; <stmt>; <stmt>) { <stmts> }

const;

<var> = <expr>;   <var> < <expr>;   <var> = <expr>;   <var> = <expr>;

a   <term>;   a   <term>;   a   <term> - <term>   b

const;   const;   <var>   const   <term> + <term>;

a   <var>   <var>;

b   a;