EECS483 D9: Midterm Exam Review

Chun-Hung Hsiao

March 15, 2013

Announcements

- Project 3 due on midnight Monday
- Extra office hour during 4-5pm next Monday

Problem A1: Which are Regular?

- (O) English words in Merriam-Webster Dictionary
  - Finite set
- (X) The set of all palindromes
  - Need O(n) space to check
- (O) Strings that can be derived by "E → E + E | id"
  - id (‘+’ id)*
- (O) Strings that can be derived by "E → E + E | (E) | id" in at most 100 derivation steps
  - Finite set
- (X) All regular expression strings
  - Need to match parentheses

Problem A2: Which are True?

- (O) Any ambiguous CFG is not LR(1)
- (X) Any unambiguous CFG is LR(1)
- (X) An unambiguous CFG can only have a unique derivation for each string in its language
  - Unique leftmost/rightmost derivation
- (O) If a CFG contains a shift/reduce conflict in LALR (1) parsing, then it would also contain a shift/reduce conflict in LR(1) parsing
  - LALR(1) doesn't introduce new s/r conflict
- (X) Language of an unambiguous CFG is regular

Problem B1

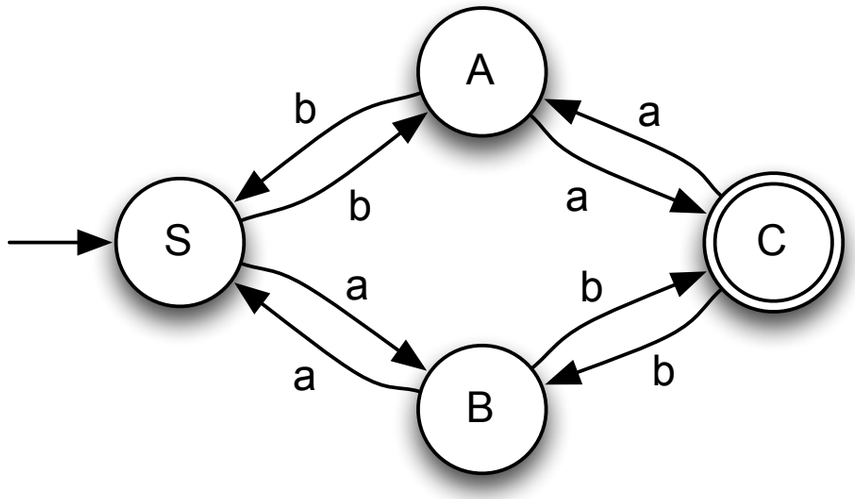
- ```
  (ab|b)+       { printf("1"); }
  [ab]?ab       { printf("2"); }
  (aa)*a?b      { printf("3"); }
  .             { printf("4"); }
  ```
- `abab aaab abbab aab ab`
  ```

Problem B2

- Counter example of `/\*.*\*/`
  - Does not match newlines
  - Do longest match so won't terminate at the first appearance of '*/'

Problem B3

- S → aB | bA
  A → aC | bS
  B → bC | aS
  C → aA | bB | ε



- A : strings with odd ‘a’s and even ‘b’s
  B : strings with even ‘a’s and odd ‘b’s
  C : strings with even ‘a’s and ‘b’s
  S : strings with odd ‘a’s and ‘b’s

- (aa|bb)*((ab|ba)(aa|bb)*(ab|ba)(aa|bb)*)* (ab|ba)(aa|bb)*

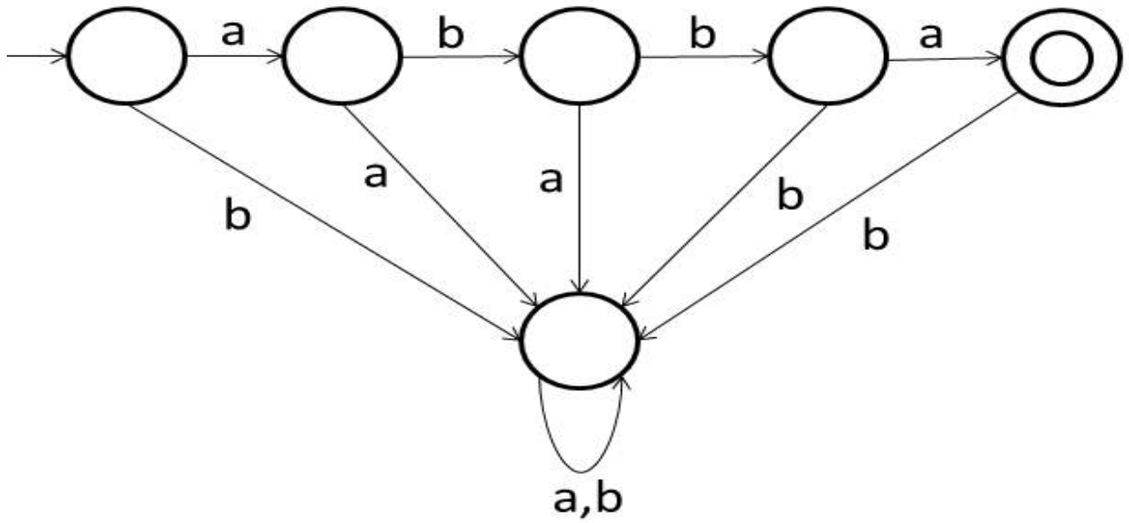
Problem B4

- L is regular, S ⊆ L, then is S regular?
  - L = [ab]*, S = balanced “a-b parentheses”

Problem B5

- L = { x | x ∈ $\sum^*$, x mod 3 = 0 when interpreted as a number in binary }
  - Given any binary string 101110
  - $(101110)_2 = 1\times2^5 + 0\times2^4 + 1\times2^3 + 1\times2^2 + 1\times2^1 + 1\times2^0$
  - next remainder := current remainder × 2 + next digit

Problem B6

- 

- To make the complement of L
  - Add an erroneous state to recover all missing transitions
  - Swap the set of accepting states and non-accepting states

Problem B7

- All strings with at most two occurrences of the substring 00
- (1|01)* ((001(1|01)*00 | 000 | 00) (1|10)*)?

Problem B8

-

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

- Both A::Set() and B::Set() access the global x
- The Set() calls in A::SetAndPrint() and C::SetAndPrint() are dynamically dispatched
- C::SetAndPrint() prints out C::x, not the global x

Problem B8

```
void main() {
     A a1;
     A a2;
     a1 = New(B);
     a2 = New(C);
     {
          int x;
          x = 5;
          SetAndPrint();
          Print(x);
     }
     a1.SetAndPrint();
     a2.SetAndPrint();
     Print(x);
}
```

set and print global x := 2

print local x = 5

A::SetAndPrint() calls B::Set(2):
set and printglobal x := 5

C::SetAndPrint() calls A::Set(4):
set global x := 6 and print C::x = 0

print global x = 6

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;



    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;


    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
class A {

    void Set(int y) {
        x = y + 2;
    }
    void SetAndPrint(){
        Set(2);
        Print(x);
    }
}
```

```
class B extends A {

    void Set(int y){
        x = y + 3;
    }
}
```

```
class C extends A {
    int x;


    void SetAndPrint() {
        Set(4);
        Print(x);
    }
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

vtable for Problem B8

```
void main() {
     A a1;
     A a2;
     a1 = New(B);
     a2 = New(C);
     {
          int x;
          x = 5;
          SetAndPrint();
          Print(x);
     }
     a1.SetAndPrint();
     a2.SetAndPrint();
     Print(x);
}
```

A::vtable

Index	Function Pointer
0	
1	

B::vtable

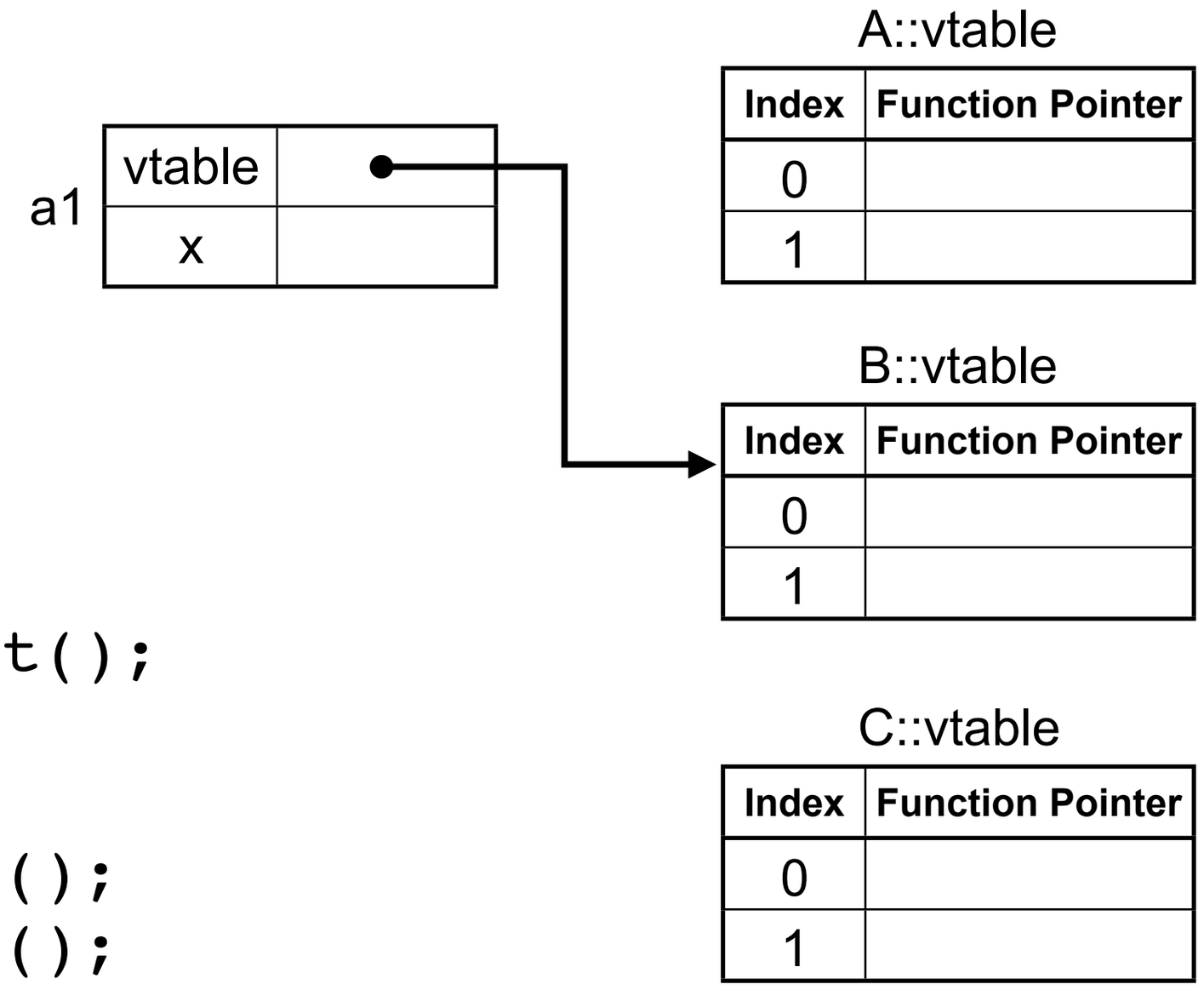
Index	Function Pointer
0	
1	

C::vtable

Index	Function Pointer
0	
1	

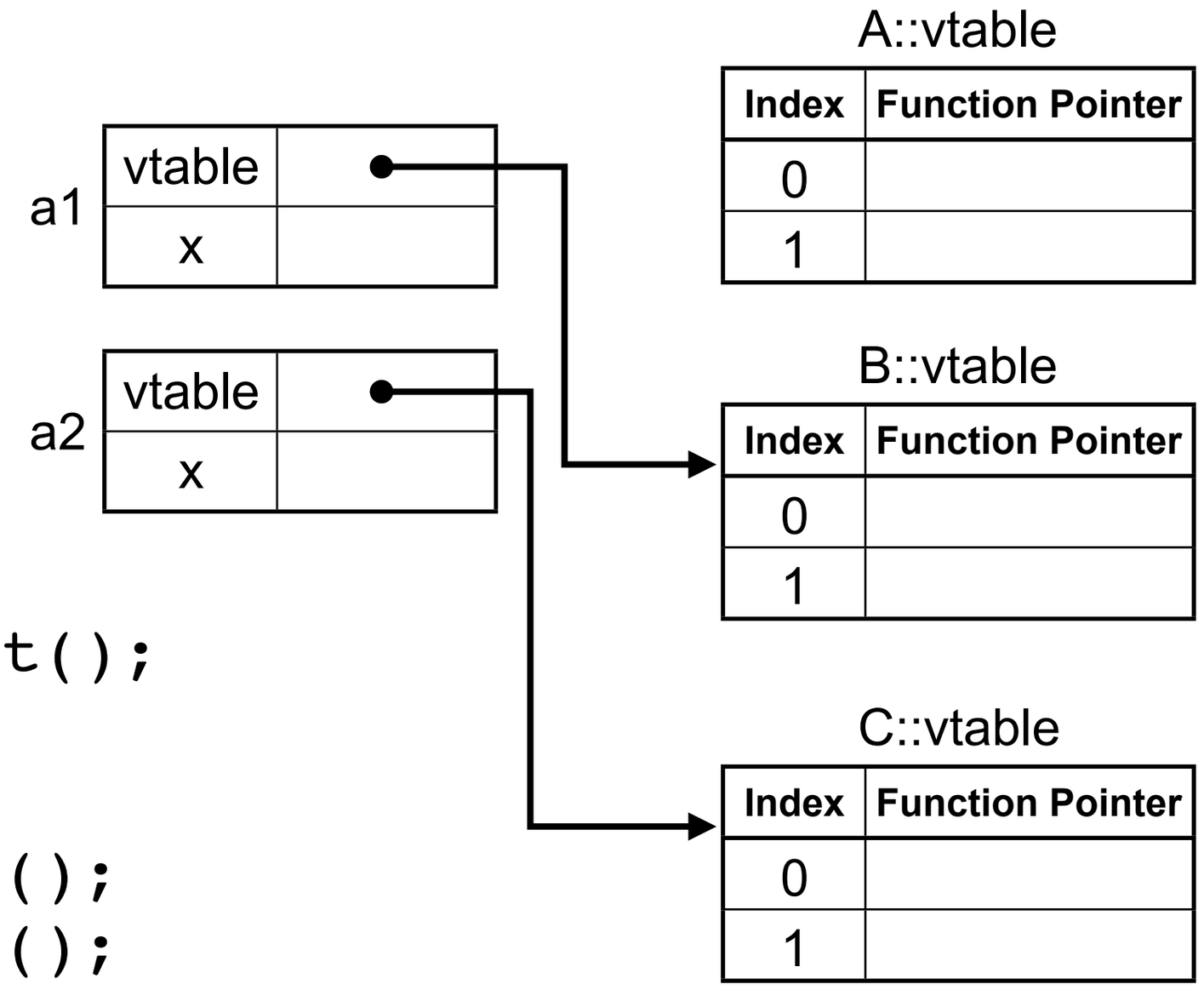
vtable for Problem B8

```
void main() {
      A a1;
      A a2;
      a1 = New(B);
      a2 = New(C);
      {
            int x;
            x = 5;
            SetAndPrint();
            Print(x);
      }
      a1.SetAndPrint();
      a2.SetAndPrint();
      Print(x);
}
```



vtable for Problem B8

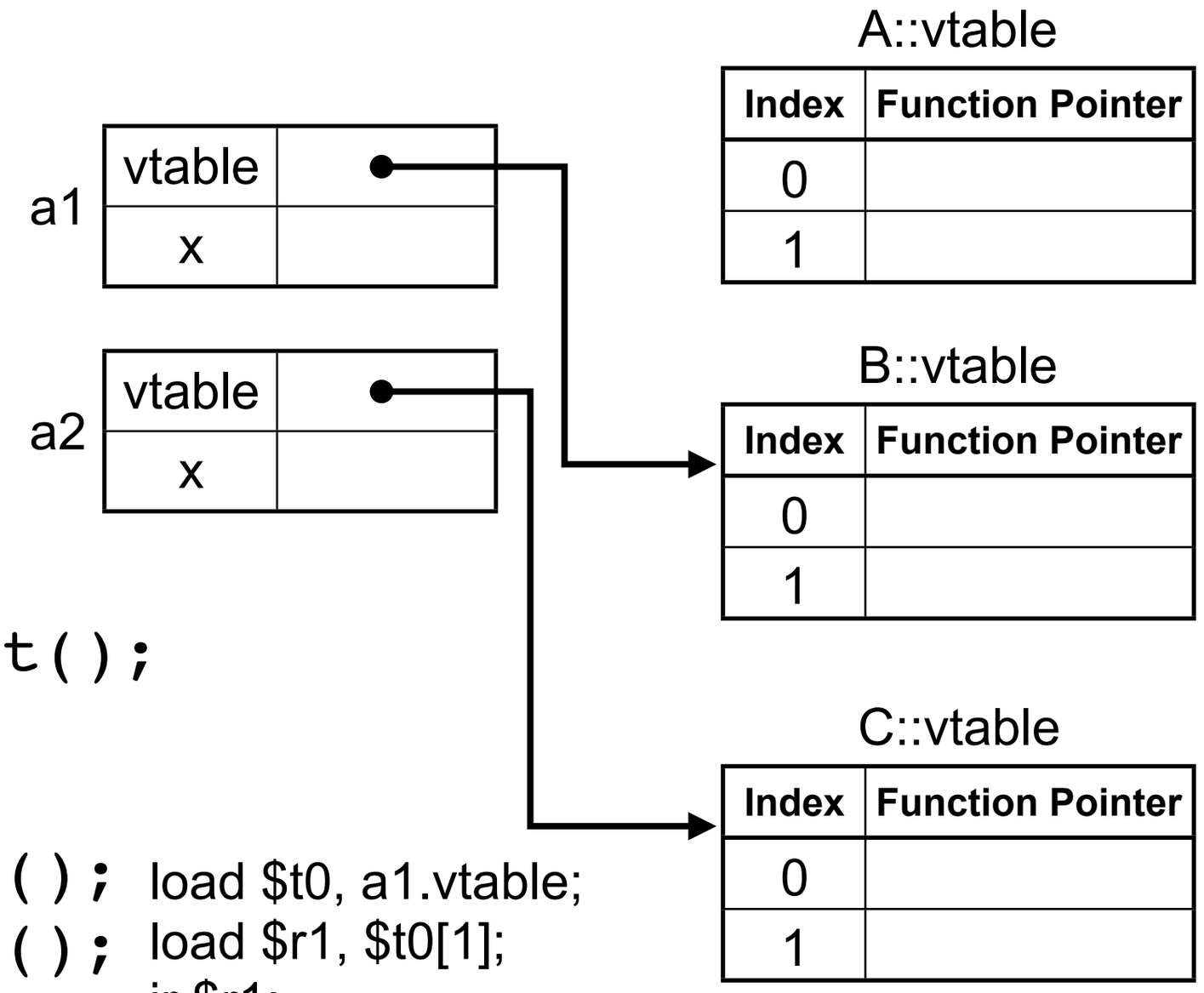
```
void main() {
      A a1;
      A a2;
      a1 = New(B);
      a2 = New(C);
      {
            int x;
            x = 5;
            SetAndPrint();
            Print(x);
      }
      a1.SetAndPrint();
      a2.SetAndPrint();
      Print(x);
}
```



vtable for Problem B8

```
void main() {
      A a1;
      A a2;
      a1 = New(B);
      a2 = New(C);
      {
            int x;
            x = 5;
            SetAndPrint();
            Print(x);
      }
      a1.SetAndPrint();
      a2.SetAndPrint();
      Print(x);
}
```

```
load $t0, a1.vtable;
load $r1, $t0[1];
jr $r1;
```



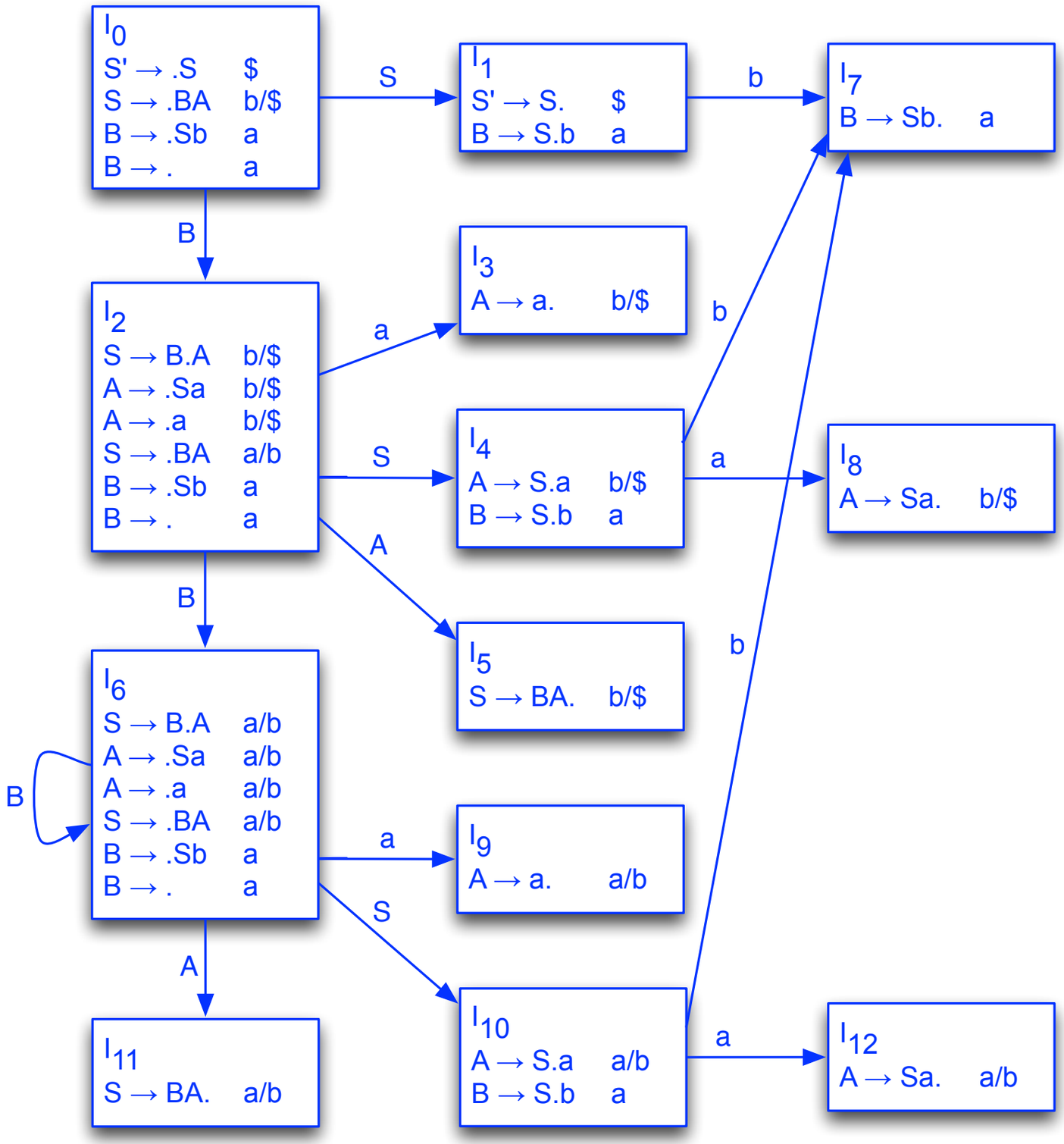
Problem C1

- S → BA
  A → Sa | a
  B → Sb | ε

- Remove the ε-probuction:
  S → BA | A
  A → Sa | a
  B → Sb

- S → BA | A
  A → BAaA' | aA'
  A' → aA' | ε
  B → aA'bB'
  B' → AbB' | AaA'bB' | ε

Problem C1


I0
S' → .S $
S → .BA b/$
B → .Sb a
B → . a
S
I1
S' → S. $
B → S.b a
b
I7
B → Sb. a
B
I2
S → B.A b/$
A → .Sa b/$
A → .a b/$
S → .BA a/b
B → .Sb a
B → . a
a
I3
A → a. b/$
b
S
I4
A → S.a b/$
B → S.b a
a
I8
A → Sa. b/$
A
I5
S → BA. b/$
B
I6
S → B.A a/b
A → .Sa a/b
A → .a a/b
S → .BA a/b
B → .Sb a
B → . a
B
a
I9
A → a. a/b
S
A
I11
S → BA. a/b
I10
A → S.a a/b
B → S.b a
a
I12
A → Sa. a/b
b

Problem C2

- R → R '|' R (alternation operation)
  R → R R (concatenation operation)
  R → R '*' (repetition)
  R → '(' R ')'
  R → a | b

- R → RR → <u>RR</u>R →* aaa
  R → RR → aR → a<u>RR</u> →* aaa

Problem C3

- S → AaAb I BbBa
  A → ε
  B → ε
- L(S → AaAb) = {a}, L(S → BbBa) = {b} => LL(1)
- FOLLOW(A) = FOLLOW(B) = {a,b}
  - Don’t know whether to reduce ε to A or B

Thanks & all the best!