44-525 Theory and Implementation of Programming Languages

Exam 2 Fall 2012

1. (3 pts) The language {0n1 : n ≥ 0} is (circle all correct answers)

(a) regular

(b) context-free

(c) recognized by a deterministic finite automaton

2. (3 pts) If a language L is recognized by a pushdown automaton, then it is also recognized by a deterministic finite automaton.

(a) true

(b) false

3. (3 pts) If a language L is context-free, then there is a pushdown automaton that recognizes L.

(a) true

(b) false

4. (10 pts) Write a context-free grammar for the language consisting of all nonempty strings of 0’s having even length.

5. (5 pts) Let G be the grammar below.

S → aSd | T

T → bTc | e

Find five strings in L(G), all of length ≤ 5.

6. (10 pts) Let G be the grammar below.

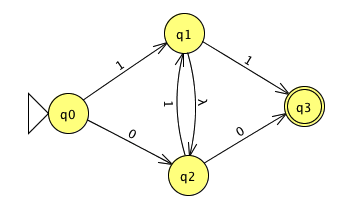
S → aSd | T

T → bT | Tcc | x

Write a derivation of the string abxccd. Apply exactly one production at each step.

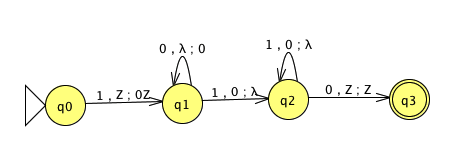
7. (10 pts) Let G be the grammar in problem 6. Draw a parse tree for the string aabbxccdd.

8. (8 pts) Find all strings accepted by the non-deterministic finite automaton below.



9. (10 pts) Design a nondeterministic finite automaton that accepts the language of all nonempty strings of 0’s whose length is a multiple of 3. For example, 000 and 000000 are accepted but 0000 is not accepted.

10. (8 pts) Let P be the pushdown automaton shown below.



Circle all of the strings that are accepted by P.

110

1100

1110

10110

10001110

1001110

1010

100110

11. (10 pts) Assume we are using the following grammar for arithmetic expressions:

0. E’ → E

1. E → E+T

2. E → T

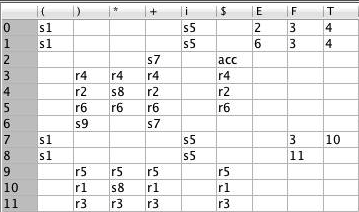
3. T → T\*F

4. T → F

5. F → (E)

6. F → i

Here is a parsing table for this grammar:



Assume the input string is i+i+i. Trace the execution of the LR parser for this input by filling in the rows of the table below. Stop when the action is acc.

|  |  |  |
| --- | --- | --- |
| **Stack** | **Remaining Input** | **Action** |
| 0 | i+i+i$ |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

12. (10 pts) Find the output of the PL0 program below.

JUMP 0 1

OUTPUT

ALLOC 0 5

PUSH 0 3

STORE 0 3

PUSH 0 1

STORE 0 4

LOAD 0 3

PUSH 0 10

SLT 0 0

BZ 0 21

LOAD 0 4

LOAD 0 3

ADD 0 0

STORE 0 4

LOAD 0 3

PUSH 0 2

ADD 0 0

STORE 0 3

LOAD 0 4

OUT 0 0

JUMP 0 6

LOAD 0 4

OUT 0 0

RET 0 0

13. (10 pts) Assume variables a, b, and c are stored at offsets 3, 4, and 5, respectively. Write stack machine code that implements this PL0 code:

if (a > b + c)

{

c = 10 \* (a + b + 3 \* c) + 4;

}

write(c);

Work Space (DO NOT REMOVE THIS PAGE)

Work Space (DO NOT REMOVE THIS PAGE)