

CMSC330 - SUMMER 2006 - MIDTERM 1

INSTRUCTOR: GUILHERME FONSECA

- Write all answers in the answers book provided.
- You can keep the exam. Only return the answers book.
- You are allowed to consult one letter-size paper, handwritten on one side. Besides that, the exam is closed book, closed notes.
- There are 8 question, totaling 110 points, in this exam. You have 1 hour and 20 minutes to finish it.

(1) (20 points) What is the output of each Ruby program below? Ignore any possible warning message.

(a)

```
puts("ab" +  
    if nil  
      "cd"  
    else  
      "ef"  
    end)
```

(b)

```
a = b = ["c", "a", "b"]  
a = a.sort  
puts b
```

(c)

```
a = [1, 2, 3]  
b = ["x", "y"]  
c = [a, b, [a, b]]  
puts c[-1][0]
```

(d)

```
h = Hash.new(0)  
h["a"] = h["b"]  
h["b"] = 7  
h["c"] += 2  
puts "#{h["a"]} #{h["b"]} #{h["c"]}"
```

(2) (10 points) Write a Ruby program that reads several lines from the input, and prints only the lines that contain **exclusively** the following characters: uppercase and lowercase letters, digits, and underscore. For example, lines that contain space or punctuation should not be printed.

(3) (21 points) Write a formal regular expression for each of the languages below. The alphabet is $\Sigma = \{a, b\}$. The only operators allowed are concatenation, $*$ and $|$ (do not write a Ruby regular expression).

- (a) $\{w \mid w \text{ begins with } a \text{ and ends with } a\}$
- (b) $\{w \mid \text{all } a\text{'s are immediately followed by } b \text{ in } w\}$
- (c) The union of the two languages above.

- (4) (8 points) Write a formal regular expression the language below. The alphabet is $\Sigma = \{a, b, c\}$. The only operators allowed are concatenation, $*$ and $|$ (do not write a Ruby regular expression).

$\{w \mid \text{all consonants are adjacent to a consonant on at least one side in } w\}$

Notice that b and c are the only consonants in the alphabet Σ . For example, a , bb , bc , and $aabbabcba$ are in the language, but b , ab , and aba are not.

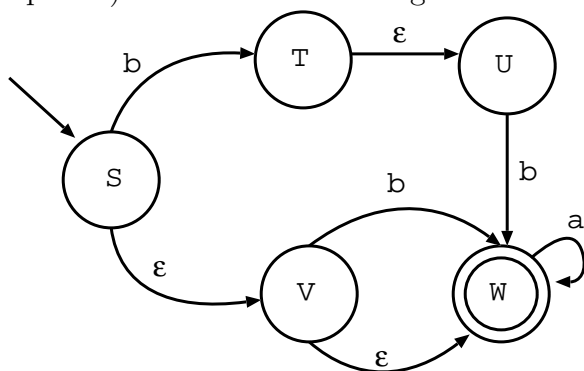
- (5) (21 points) Write a DFA for each of the languages below. The alphabet is $\Sigma = \{a, b\}$.

(a) $\{w \mid w \text{ contains at most one } b\}$

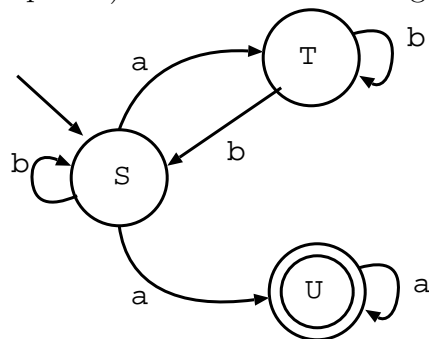
(b) $\{w \mid \#a(w) = 0 \pmod{2} \text{ and } \#b(w) \neq 0 \pmod{3}\}$

(c) $\{w \mid w \text{ ends with } aab\}$

- (6) (10 points) Convert the following NFA to a NFA without ε transitions.



- (7) (10 points) Convert the following NFA without ε transitions to a DFA.



- (8) (10 points) Convert the following formal regular expression to a NFA:

$$((aa|bb|c)^*cc(a|b))^*$$