

# CPTS 355 Sample Midterm 1

1) [15 pts] **Directions:** Short answer – each answer is an expression or a couple of sentences at most.. (3 points for each subproblem)

a) In Scheme what is `(let ((L '(1 2) (3 4))) (cons (caar L) (caddr L)))`?

b) What is the value of the following postfix (RPN) expression? `4 4 * 4 32 chs * - +`  
(chs means “change sign” on the top stack element)

c) Describe a Turing machine. Draw a picture labeling the key components?

d) Suppose that while a Turing machine is executing you observe it return to a state, tape contents and head position that you have seen before during the execution. What can you conclude about the TM’s future behavior? Why?

e) Why is Turing completeness an important property for a programming language to have? Describe one way that you could demonstrate that a particular PL is Turing complete (provided that you had a computer with infinite memory).

2a) [10 points] Describe what occurs when each **def**, **begin**, **end**, **sub**, and **add** operation is executed in the following PostScript program. You **must** list the operations **in the order they are executed**. For each one say which line number it occurs on and what operands it uses. The first two are done for you as an example. (Of course, the line numbers are not part of the program. There may be more blank lines than required answers.)

```

Line
0      /y 3 def /x 2 def
1      /f { dup dict begin 1 sub /y exch def
2          /x 4 def
3          x y add
4          /y exch def
5      } def
6      y f y

```

Line	Operation	Operands
<u>0</u>	<u>def</u>	<u>/y 3</u>
<u>0</u>	<u>def</u>	<u>/x 2</u>
—	—	—
—	—	—
—	—	—
—	—	—
—	—	—
—	—	—
—	—	—
—	—	—
—	—	—

2b) [5 pts] What value(s) are on the operand stack when the program finishes execution? Circle the value that is at the top of the operand stack.

2c) [5 pts] What is in the dictionary at the top of the dictionary stack when the program finishes execution?

3a) [10 points] Draw a derivation tree for the sentence

$$4 * 5 + 2 / 3$$

using the grammar given below. Assume that  $\langle e \rangle$  is the start symbol.

$\langle e \rangle \rightarrow \langle e \rangle \langle \text{mulop} \rangle \langle t \rangle \mid \langle t \rangle$   
 $\langle \text{mulop} \rangle \rightarrow * \mid /$   
 $\langle t \rangle \rightarrow \langle f \rangle \langle \text{addop} \rangle \langle t \rangle \mid \langle f \rangle$   
 $\langle \text{addop} \rangle \rightarrow + \mid -$   
 $\langle f \rangle \rightarrow 2 \mid 3 \mid 4 \mid 5$

3b) [5 pts] Using the tree you have drawn above determine the value of the expression  $4 * 5 + 2 / 3$  according to this grammar. Assume that  $/$  is integer division with the quotient truncated toward 0.

3c) [5 pts] In the above grammar the multiplication operators are \_\_\_\_\_ associative and the addition operators are \_\_\_\_\_ associative.

4a) **[15 pts]** Define a recursive Scheme function, `countsubs`, that will take a list of lists as input and produce as output a list whose elements are the number of elements in each sublist. For example, `(countsubs '( (1 2 4) (14 5 6 7) () (1) ))` should produce `(3 4 0 1)` as its answer. You may assume that each element of the argument list is in fact a list – you don't have to check for bad input. You will want to define an auxiliary function to count the elements of a sublist. In your definitions you may use only `cons`, `car`, `cdr`, `+`, `null?`, `if`, `cond`, `define`.

```
(define (countsubs L) (
```

```
))
```

4b) **[5 pts]** Pairprods can also be defined using the built-in higher-order function `map`. Define `countsubs` using `map`.

```
(define (countsubs L)
  (map
```

```
    )
  ))
```

5a) **[12 pts]** Python has at least 3 types of sequences, *lists*, *strings*, and *tuples*. For each one tell whether a value of that type is *mutable*, or *immutable*.

Lists are \_\_\_\_\_

Strings are \_\_\_\_\_

Tuples are \_\_\_\_\_

What does immutable mean?

5b) **[8 pts]** Write a python function that will return a new string containing the same characters as string S but sorted in ascending order. Look at 5c below before answering.

```
def strsort(S):
```

```
    return S2
```

5c) **[5 pts]** One might be tempted to write

```
def strsort(S):  
    return S.sort()
```

What are two reasons this would not work?