

CS241e Review Questions

Note that these questions have been prepared by someone who has not seen the final exam, and therefore may or may not be representative of questions that are on the actual exam.

1. What is the output of scanning?

2. Sort the following in the order which they are used to turn a Lacs program into machine code:
 - Parsing
 - Code generation
 - Assembling
 - Type checking
 - Scanning

3. T/F: Lacs is a context-free language

4. T/F: if a DFA has n states and accepts a string of length n , then it accepts an infinite number of strings.

5. T/F: Every set of finite strings is determined by a unique DFA

6. T/F: There exists some $k \geq 2$ such that there is a NFA with k states but no equivalent DFA with k states.

7. Show that the following grammar is ambiguous:
 - $S \rightarrow AB$
 - $A \rightarrow AB$
 - $B \rightarrow BB$
 - $B \rightarrow BA$
 - $A \rightarrow x$

$B \rightarrow y$

8. Create a grammar that accepts the set of balanced parentheses.
9. Give an example of a language which is context-free but not regular.
10. Is the set of all palindromes over binary strings (i.e. binary strings that are read the same forwards as backwards) a context-free language?
11. What is the difference between a closure and a normal function?
12. Why can't closures be implemented using a simple 1-1 mapping to machine code, in the same way that a function like `ifstmt` can be?
13. During which phase of compilation does the following Lacs programs throw an error?

```
def main(a: Int, b: Int): Int = {  
    a++b  
}
```

14. During which phase of compilation does the following Lacs programs throw an error?

```
def main(a: Int, b: Int): Int = {  
    $a*$b  
}
```

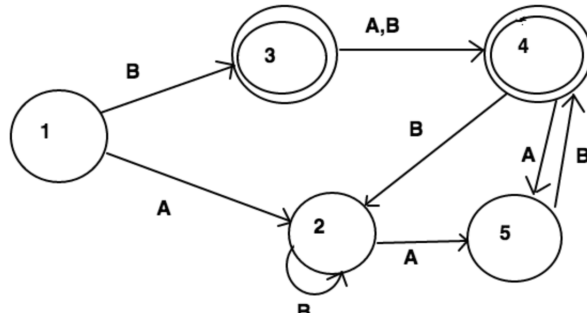
15. During which phase of compilation does the following Lacs programs throw an error?

```
def main(a: (Int) => Int, b: Int): (Int) => Int = {
  a(b)
}
```

16. In this course we used variable sized blocks of memory, however you can also implement memory using fixed-sized blocks. For example, we could decide to only give out multiples of 16 words of memory. So if 32 words of memory was requested, we would give two 16 word blocks of memory, and if 4 words of memory was requested, we would give one 16 word block of memory. What memory related problem does this solve? Does it create any new problems?

17. Explain what the following code does:
 $\lambda.x\ x\ \lambda.y\ y\ \lambda.a.b\ a$

18. Give the output of the maximal munch algorithm for the following DFA with the input "bababbabb"



(Note that there is "B" is a transition from the state 2 back to the state 2, it got cut off)