

page 1	page 2	page 3	page 4	page 5	Total / 54

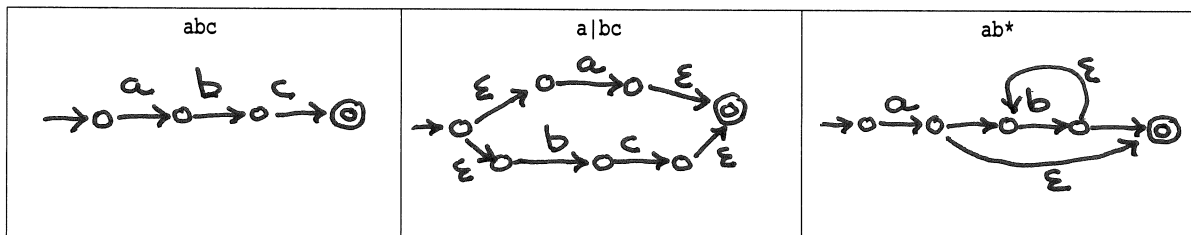
Please print clearly:

Name: **SOLUTION**

Login: @ucsc.edu

No books; No calculator; No computer; No email; No internet; No notes; No phone. Neatness counts! Do your scratch work elsewhere and enter only your final answer into the spaces provided. Points will be deducted for messy or unreadable answers.

1. Using Thompson's construction exactly, draw nondeterministic finite αὐτόματα for each of the following regular expressions. [3✓]



2. Write a **bison** grammar for the following language. Clearly separate the part 1 declarations from the part 2 grammar by indicating the position of the `%`. Do not show semantic actions. [5✓]

- A program is a sequence of zero or more statements.
- A statement is an assignment or a query.
- A statement is an **IDENT**, then equal (=) sign, then an expression, and ends with a semi-colon (;).
- A query is an **IDENT** followed by a question mark (?).
- An expression is a sequence of **IDENT**s and **NUMBERS** connected with the binary operators +, -, \*, and /, with the same precedence and associativity as C. Parentheses are allowed as in C. Make the expression grammar ambiguous and resolve it in the usual **bison** way.

```
%token IDENT NUMBER
%left '+' '-'
%left '*' '/'
%start prog
%%
prog: prog stmt | ;
stmt: asgt | query;
query: IDENT '?';
asgt: IDENT '=' expr;
expr: expr '+' expr | expr '-' expr
    | expr '*' expr | expr '/' expr
    | '(' expr ')' | IDENT | NUMBER;
```

3. Write **flex** patterns for the following: [2✓]

- An **IDENT** is a sequence of one or more upper- or lower-case letters and digits, but may not begin with a digit. Also, underscores may appear between any consecutive letters or digits, but may not begin or end an **IDENT** or appear next to another underscore.

`[A-Za-z](_?[A-Za-z0-9])*`

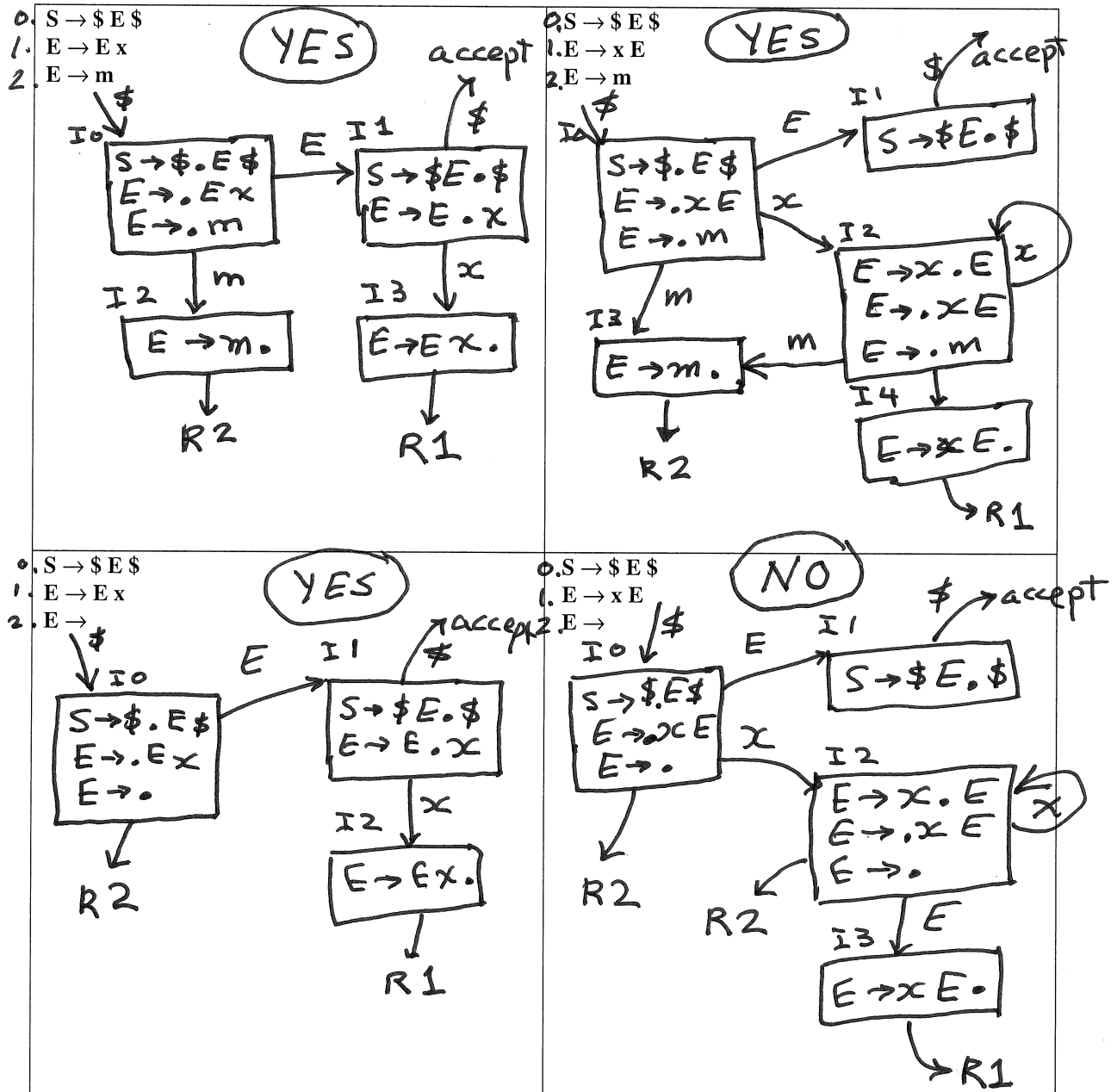
- A **NUMBER** is a sequence of decimal digits, possibly with a decimal point. But the decimal point may not appear as the first or last character. An optional exponent follows, which begins with the upper- or lower-case letter E, followed by an optional + or - sign, and followed by one or more digits.

`[0-9]+(\.[0-9]+([Ee][+-]?[0-9]+)?)?`

4. After a call instruction, on entry to a function, what machine instructions are executed to save the frame pointer and allocate enough space for  $N$  bytes on the stack? What instructions are executed just prior to a return instruction to remove the local stack frame from the function call stack? The frame pointer is  $\%rbp$  and the stack pointer is  $\%rsp$ . The `movq` instruction moves the value of its first operand into its second operand. The `subq` instruction subtracts its first operand from its second. [2v]

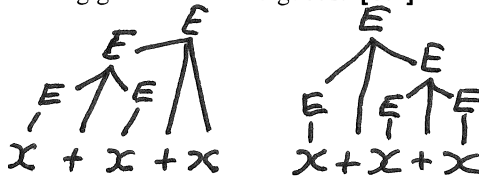
Code on entry to the function (prolog)	Code just before the return instruction (epilog)
<pre>pushq %rbp movq %rsp, %rbp subq \$N, %rsp</pre>	<pre>movq %rbp, %rsp popq %rbp ret</pre>

5. For each of the following grammars, in the style of the LR handout discussed in class, draw the characteristic finite state machine (CFSM). Draw your diagrams neatly *completely inside* each box. A messy or unreadable diagram will lose points. Also, write the word **YES** if the grammar is LR(0), and **NO** if it is not LR(0). [8v]



6. Show that the following grammar is ambiguous. [1✓]

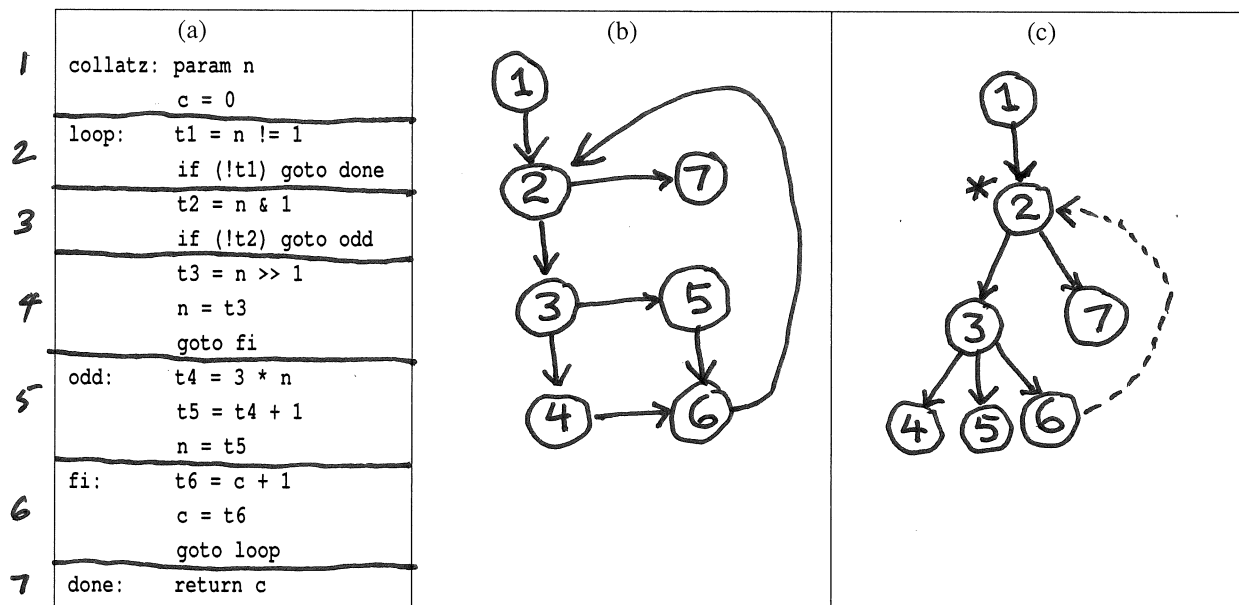
$E \rightarrow E + E$   
 $E \rightarrow E * E$   
 $E \rightarrow (E)$   
 $E \rightarrow x$



same input can generate two different parse trees

7. Basic blocks.

- (a) Given the code shown in the box, draw a horizontal line immediately above the leader of each basic block, thus separating it from the last instruction in the preceding basic block. Number the basic blocks in sequence as 1, 2, ..., etc. in the same order as the instructions appear. [1✓]
- (b) Draw a data flow diagram with each circle in the diagram having the number of a basic block, and with arrow showing flow of control. Draw a follow link with a downward arrow and a branch link with an arrow that starts out from the side of the source block. [2✓]
- (c) Draw the dominator tree with solid downward arrows showing the **dom** relationship. The root should be at the top of the diagram. Draw a dotted arrow showing the back edge. Write an asterisk next to the head of the natural loop. [2✓]

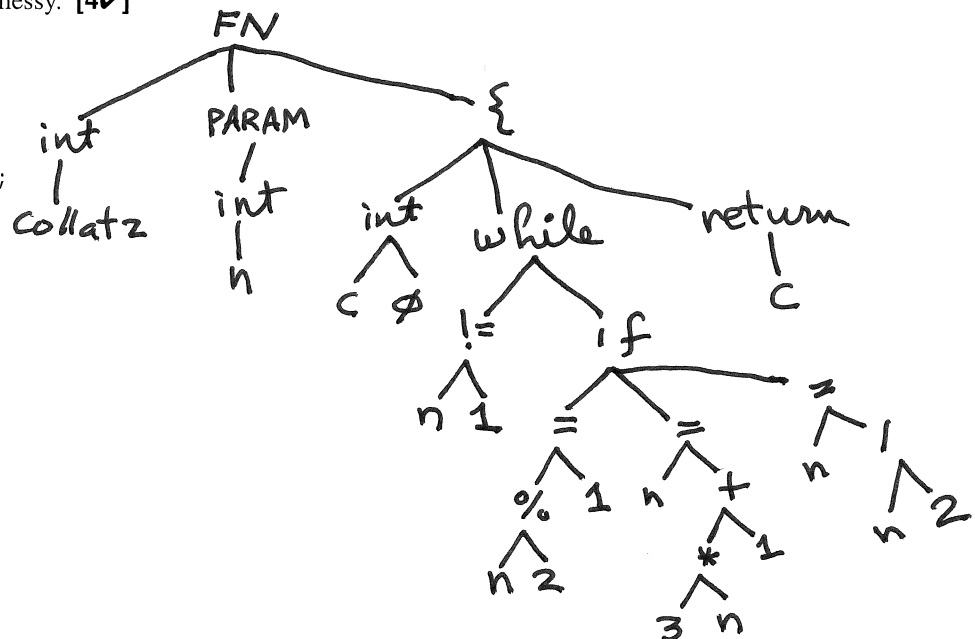


8. Draw an abstract syntax tree for the following function, using the specifications for the project. Draw it neatly or lose points for being messy. [4✓]

```

int collatz (int n) {
    int c = 0;
    while (n != 1)
        if (n % 2 == 1)
            n = 3 * n + 1;
        else
            n = n / 2;
    return c;
}

```



Multiple choice. To the *left* of each question, write the letter that indicates your answer. Write **Z** if you don't want to risk a wrong answer. Wrong answers are worth negative points. [12✓]

number of correct answers		$\times 1 =$	$= a$
number of wrong answers		$\times \frac{1}{2} =$	$= b$
number of missing answers		$\times 0 =$	$0$
column total	12		$= c$
$c = \max(a - b, 0)$			

1. Which memory management system, when it is finished running, has heap data most tightly packed together?

(A) copying collection with semispaces.  
(B) `malloc(3)` and `free(3)`.  
(C) mark and sweep collection.  
(D) reference counted memory management.

2. What will the following grammar recognize?

$A \rightarrow Ax$

$A \rightarrow y$

(A) xxxxxxxxy  
(B) xxxxxxxyy  
(C) yxxxxxxxx  
(D) yyyyyyyxx

3. Uninitialized variables is a \_\_\_\_\_ problem.

(A) basic block  
(B) control flow  
(C) data flow  
(D) dominator

4. The first instruction in the prolog of a function is usually:

(A) `movq %rsp, %rbp`  
(B) `popq %rbp`  
(C) `pushq %rbp`  
(D) `pushq %rsp`

5. Which describes what is **NOT** the leader of a basic block?

(A) the first instruction after a call  
(B) the first instruction after a jump  
(C) the first instruction after a return  
(D) the target of a jump

6. If parameters are passed on the stack, as is the case for the x86 32-bit address space, what might be a reasonable address of one of the functions incoming arguments after the local frame has been allocated?

(A) `+16(%ebp)`  
(B) `+16(%esp)`  
(C) `-16(%ebp)`  
(D) `-16(%esp)`

7. The subset construction converts:

(A) a DFA into a regular expression  
(B) a regular expression into a DFA  
(C) a regular expression into a NFA  
(D) an NFA into a DFA

8. An abstract syntax tree:

(A) has all operators as interior nodes.  
(B) has all operators as leaf nodes.  
(C) has multiple root nodes.  
(D) is constructed by the function `yylex`.

9. In the mark and sweep collection algorithm, the mark phase

(A) scans all return addresses on the stack.  
(B) scans everything on the heap.  
(C) scans only static variables.  
(D) scans the entire root set.

10. A garbage collector frees and recycles all \_\_\_\_\_ heap data.

(A) dead  
(B) live  
(C) reachable  
(D) unreachable

11. If the function `g` is nested inside the function `f`, then `g` can access variables local to `f` but not local to `g` by using its:

(A) access link  
(B) control link  
(C) return address  
(D) stack pointer

12. Non-static variables in C which are local to a function are assigned their addresses:

(A) at compile time.  
(B) at link time.  
(C) when the function is called.  
(D) when the program is loaded into memory.

Multiple choice. To the *left* of each question, write the letter that indicates your answer. Write **Z** if you don't want to risk a wrong answer. Wrong answers are worth negative points. [12✓]

number of correct answers		$\times 1 =$	$= a$
number of wrong answers		$\times \frac{1}{2} =$	$= b$
number of missing answers		$\times 0 =$	$0$
column total	12		$= c$
$c = \max(a - b, 0)$			

1. The root set of a garbage collector is :
- (A) All of the static and local variables pointing at the heap.
- (B) All shared pointers used for reference counting memory management.
- (C) The root of the abstract syntax tree preserved in the ELF symbol table.
- (D) The root of the tree of symbol tables used to find variable names.

2. Static variables are assigned their addresses :

- (A) at compile time.
- (B) at link time.
- (C) when the function is called.
- (D) when the program is loaded into memory.

3. Class files generated for the Java virtual machine contain what kind of code ?

- (A) an abstract syntax tree
- (B) stack machine code
- (C) three-address code
- (D) x86\_64 machine instructions

4. A depth-first spanning tree is used to construct what ?

- (A) a non-deterministic finite αὐτόματον.
- (B) the abstract syntax tree.
- (C) the dominator tree of a function.
- (D) the graph of basic blocks.

5. The confluence operator in an any-path (either forward or backward) data flow problem is :

- (A)  $\cup$
- (B)  $\cap$
- (C)  $\prod$
- (D)  $\Sigma$

6. The string yxxxxxxx is a sentence in the language defined by which grammar ?

- (A)  $A \rightarrow A x$   
 $A \rightarrow y$
- (B)  $A \rightarrow A y$   
 $A \rightarrow x$
- (C)  $A \rightarrow x A$   
 $A \rightarrow y$
- (D)  $A \rightarrow y A$   
 $A \rightarrow x$

7. The grammar :

- $A \rightarrow x$   
 $A \rightarrow y$

- (A) is both LR(0) and SLR(1).
- (B) is LR(0) but not SLR(1).
- (C) is not LR(0) but is SLR(1).
- (D) is neither LR(0) nor SLR(1).

8. Which of the following items will cause a reduction action to be added to the state ?

- (A)  $E \rightarrow \bullet E + T$
- (B)  $E \rightarrow E \bullet + T$
- (C)  $E \rightarrow E + \bullet T$
- (D)  $E \rightarrow E + T \bullet$

9. Which of the following items was entered into the state during a closure operation ?

- (A)  $E \rightarrow \bullet E + T$
- (B)  $E \rightarrow E \bullet + T$
- (C)  $E \rightarrow E + \bullet T$
- (D)  $E \rightarrow E + T \bullet$

10. For object oriented code generation with single inheritance only, what is the expected space overhead ?

- (A) one pointer per class table
- (B) one pointer per instance field
- (C) one pointer per allocated object
- (D) one pointer per virtual function

11. On the x86\_64, what is the alignment requirement (in bytes) for %rsp and %rbp ?

- (A) 0x10
- (B) 0x1000
- (C) 0x100000
- (D) 0x10000000

12. What is the possible output from the following program ?

```
void main() {printf("%p\n", main); }
```

- (A) (null)
- (B) 0x4007e6
- (C) 4196326
- (D) UHÅæ@

