

Undergraduate Exam

CSE-304 Compiler Design

Fall '97

Final Exam

December 16, 1997.

Duration: 2 hours, 45 minutes.

INSTRUCTIONS

Read the following carefully before answering any question.

- The exam for graduate students is different from the exam for the undergraduate students. Make sure you are reading the correct question paper.
- Doubts about the questions will be answered only in the first 15 minutes of the exam. So, read the questions carefully at the beginning of the exam.
- Keep your answers brief and precise.
- Answer each question on a separate sheet of paper. Write your name **and** your id number on the first sheet. Write your id number on the top right hand corner of each additional sheet.
- There are 8 questions in all, in 4 pages, for a total of 105 points. The maximum score is 100.

GOOD LUCK!

1. [Total: 10 points]

- (a) [5 points] Write a `Lex` specification to convert each occurrence of “++” in a C program to “+ 1”.
- (b) [5 points] Write a `Lex` specification to convert every continuous sequence of blank lines (a blank line contains only white space characters) into a single line containing “<P>”.

2. [10 points] `LaTeX` and `HTML` are two document formatting languages. We want to build a translator to convert documents structured using `LaTeX` into `HTML`. In `LaTeX`, the text to be formatted is specified between `\begin{document}` and `\end{document}` and in `HTML`, it is specified between `<html>` and `</html>`. In `LaTeX`, a list of (unnnumbered) bullet items is specified using `\begin{itemize}` and `\end{itemize}` and in `HTML`, the list is given using `` and ``. Each item in the list begins with `\item` in `LaTeX`, and with `` in `HTML`. A sample translation of a `LaTeX` document to `HTML` is shown below:

<code>LaTeX:</code>	<code>HTML:</code>
<pre>\begin{document} This is a sample latex document. A list of bullet items can be created as follows. \begin{itemize} \item Start an \itemize environment; \item Place each item in with a \item command. \end{itemize} \end{document}</pre>	<pre><html> This is a sample latex document. A list of bullet items can be created as follows. Start an itemize environment; Place each item in with a item command. </html></pre>

Describe how you will structure a system to translate `LaTeX` documents to `HTML`. Explain what tools you will use to construct such a translator, and why.

Undergraduate Exam

3. [Total: 10 points]

- (a) [5 points] Let L be the set of all bit strings (*i.e.*, strings over alphabet $\Sigma = \{0, 1\}$) that are divisible by 4. Can L be represented by a regular expression? Justify.
- (b) [5 points] Let L' be the set of all bit strings (*i.e.*, strings over alphabet $\Sigma = \{0, 1\}$) that are divisible by 3. Can L' be represented by a regular expression? Justify.

4. [Total: 20 points] Consider the following grammar G_4 :

$$\begin{aligned} E &\longrightarrow E + E \\ E &\longrightarrow E * E \\ E &\longrightarrow \mathbf{id} \end{aligned}$$

- (a) [5 points] Compute the FIRST and FOLLOW sets for E .
- (b) [5 points] Is G_4 an LL(1) grammar? Justify.
- (c) [5 points] Compute the LR(0) sets of items for G_4 .
- (d) [5 points] Which states in the SLR(1) action table for G_4 has reduce actions? On what symbols are these reduce actions done?

5. [Total: 10 points]

- (a) [5 points] Consider the following grammar G_5 that generates binary numbers representing integers:

$$\begin{aligned} S &\longrightarrow L \\ L &\longrightarrow L B \\ L &\longrightarrow B \\ B &\longrightarrow \mathbf{0} \\ B &\longrightarrow \mathbf{1} \end{aligned}$$

Let an attribute *val* give the value of the binary number generated by S . For example, on input **110**, $S.val = 6$. Determine $S.val$ using syntax-directed definitions.

- (b) [5 points] Is *val* an inherited attribute of S ? Is it an inherited attribute of L or B ? Justify.

6. [Total: 15 points]

- (a) [3 points] Explain why field and method names in `Decaf` cannot be resolved until types are known. That is, why cannot one determine which entity is represented by `a[i].x` until the type of `a[i]` is known?
- (b) [4 points] When parsing `Decaf` programs, we can encounter a variable x in some expression such that x is not declared in the current scope, nor in any enclosing scope. Explain why this is not an error.

Undergraduate Exam

(c) [8 points] Consider the following Decaf program:

```
class A {  
    public int i, j;  
}  
class B extends A {  
    private boolean i;  
    public float j;  
}  
class C extends B {  
    private boolean j;  
}  
class D extends C {  
    public float i;  
}
```

Let d be an instance of class D , and c be an instance of class C . What are the types of $c.i$, $c.j$, $d.i$ and $d.j$?

7. [Total: 15 points]

(a) [5 points] Consider the C declaration `int x[10]`; . Write the type of x using the language of type expressions given in the text (pages 345–347).

(b) [5 points] Consider the C function `strcmp`, defined as:

```
int strcmp( char *s1, char *s2)  
{  
    ...  
}
```

Write the type of `strcmp`.

(c) [5 points] Consider the function $sort(f, a)$ that sorts an integer array a and yields an integer array b using a user-specified comparison function f that compares two integers and returns a boolean value. What is the type of $sort$?

8. [Total: 15 points]

(a) [5 points] Consider the following grammar G_8 for expressions E :

$$\begin{aligned} E &\longrightarrow E + E \\ E &\longrightarrow \text{id} \\ E &\longrightarrow \text{int} \end{aligned}$$

Using a stack machine (such as the one given in Chapter 2 of the text, or Cream) as the target machine, write syntax-directed definitions to generate code for E .

(b) [5 points] Consider the grammar G'_8 derived by adding

$$E \longrightarrow E = E$$

to grammar G_8 .

Modify the syntax directed definition in (a) to include code generation for the assignment operation.

Undergraduate Exam

(c) [5 points] Consider the grammar G''_g derived by adding

$$E \longrightarrow E + = E$$

to grammar G'_g .

Modify the syntax directed definition in (b) to include code generation for expressions of the form $x += y$.

END OF EXAM