University of Toronto

CSC488S/CSC2107S Compilers and Interpreters

Winter 2015/2016

March 3, 2016.

Mid Term Test (20% of course mark)

5 questions on 2 pages. 100 marks total. 50 minutes total Open Book and Notes, Non-programmable calculators allowed, NO other electronic aids allowed Answer all questions. WRITE LEGIBLY!

If you need to make any additional assumptions to answer a question, be sure to state those assumptions in your test booklet. The line numbers on the left side of programs and grammars are for reference only and not part of the program or grammar.

1. [15 marks] The representation of information is evolving toward the use of meta languages like XML as a standard encoding format. Suppose that the text files containing source programs are now encoded in XML.

Example Program Fragment XML Encoding

if (X != 13) {	<pre><reserved>if</reserved> <parenthesis> <identifier>X</identifier></parenthesis></pre>
Y = 12 ;	<pre><operator>!=</operator> <integer>13</integer> </pre>
Z = X - 1 ;	<pre><bracket> <identifier>Y</identifier> <operator>=</operator></bracket></pre>
}	<pre><integer>12</integer> <operator>;</operator> <identifier>Z</identifier></pre>
	<pre><operator>=</operator> <identifier>X</identifier> <operator>-</operator></pre>
	<integer>1</integer> <operator>;</operator>

Describe how you would design a compiler scanner to process source programs that are encoded in this way. Emphasize what would be *different* from the traditional scanner that was discussed in lecture.

2. [20 marks] Consider the grammar

1	$V \rightarrow$	W X 'c'
2	\rightarrow	'e' Y W
3	$W \rightarrow$	'b' X Y Z
4	\rightarrow	λ
5	$X\rightarrow$	Y'a'W
6	\rightarrow	'c' 'a'
7	$Y \rightarrow $	'd' Y
8	\rightarrow	λ
9	$Z \rightarrow$	'e' V 'f'
10	\rightarrow	'c'

Where λ is the empty string, V is the goal symbol and terminal symbols are enclosed in single quotes.

a) [4 marks] is the grammar LL(1) ?

b) [16 marks] if the grammar is not LL(1) give an example of a predictor set conflict.

If the grammar is LL(1) give the predictor sets for the grammar.

3. [20 marks] Describe the symbol and type table entries that a typical compiler would make for the following declarations in a Pascal-like language.

```
1
       type R : record
 2
            en : enum( type1 , type2, type3 ) ;
            X : integer ;
 3
 4
            A: array 1.. 100 of pointer to R
 5
         end record
 6
       const PI = 3.1428 ;
7
       var X : array - 20 .. 20 of R ;
       procedure P (Y: string (10), var Z: R)
 8
 9
            var R, A: integer;
10
       end P
11
```

- **4. [20 marks]** Discuss the compiler implementation issues that arise from each of the programming language characteristics listed below.
 - a) identifiers can be up to 50 characters long.
 - b) the language uses keywords instead of reserved words
 - c) array variables can be initialized by a list of constants at the point of declaration
 - d) parsing the language requires more than one symbol lookahead
 - e) the language does not require that a variable be declared before it is used.
- **5. [25 marks]** Use the depth first structure alignment algorithm described in the lecture slides to layout the data fields in the C structure shown below.

0	typedef unsigned char boolean ;
1	struct student {
2	char name [20] ;
3	long number ;
4	boolean grad ;
5	struct {
6	short assignments[5] ;
7	boolean wroteMidterm ;
8	float midterm ;
9	boolean wroteFinal ;
10	float final ;
11	} marks ;
12	double rawMark ;
13	boolean markReported ;
14	int courseMark ;
15	};

The size and alignment factors for the basic data types:

	0							
type	align	size	type	align	size	type	align	size
	(bits)	(bits)		(bits)	(bits)		(bits)	(bits)
char	8	8	long	64	64	int	32	32
short	16	16	float	32	32	double	64	64
boolean	8	8				I		