國 立 交 通 大 學 試 題 紙

一百學年度第二次
博士班資格考

科目：編譯器設計(A)　　日期：101 年 7 月 26 日　第 1 頁 共 1 頁

請" ✓" 明　✓不可看書　可看書

* 請將答案依題號順序寫入答案卷

答題時字跡需工整，否則不予計分。Write your answers legibly; otherwise you will get zero score.

Basic questions: Please answer all three questions in this part.

1. (15 points) Find the minimal deterministic finite state machine for the following regular expression:

$$a^*(b|ac)^*ab$$

2. (15 points) Please use a context-free grammar to describe the format of a lex input file. You may ignore the portions for the C code.

3. (20 points) What are the differences between LALR(1) and SLR(1) parsers? Find a grammar that is LALR(1) but not SLR(1).

◎請用深黑色鋼筆或原子筆出題

命題老師簽名：

國 立 交 通 大 學 試 題 紙

一百學年度第二次
博士班資格考

科目：編譯器設計 (B)　　日期：101 年 7 月 26 日　第 1 頁 共 2 頁

請" ✓" 明　　✓不可看書　　可看書

* 請將答案依題號順序寫入答案卷

答題時字跡需工整，否則不予計分。Write your answers legibly; otherwise you will get zero score.

1. **(8 points)** Consider the following grammar, which generates expressions for a number of operators. Assume you have the attributes *E.min* and *E.max* that should be set to the minimum and maximum values for *E*, and an attribute **CONST**.*val* which is the value of the constant.

   $E \rightarrow$ **CONST**　{ *E.min* =?? ; *E.max* =?? ; }
   | **ID**　{ *E.min* = **ID**.*min* ; *E.max* = **ID**.*max* ; }
   | $E_1 + E_2$　{ *E.min* =?? ; *E.max* =?? ; }
   | $E_1 - E_2$　{ *E.min* =?? ; *E.max* =?? ; }
   | $E_1 * E_2$　{ *E.min* =?? ; *E.max* =?? ; }
   | (E_1)　{ *E.min* =?? ; *E.max* =?? ; }
   | $-E_1$　{ *E.min* =?? ; *E.max* =?? ; }

   Write the syntax-directed rules that calculate the range for each subexpression.

2. "Reference counting" and "mark-and-sweep" are two garbage collection algorithms. Which algorithm would work best for the following scenarios? Justify your answers.
   (a) **(3 points)** The applications will create many directed graphs where objects will refer to each other.
   (b) **(3 points)** The applications should not incur any long pause times, that is, the applications are interactive or real-time.
   (c) **(3 points)** The device will only run applications that create linked-list data structures.

3. **(5 points)** Name five different entries of an activation record. For each entry, identify its value is set before procedure invocation, during procedure execution, or right before procedure return.

4. **(5 points)** Explain the main similarity and the main difference between superscalar and VLIW architectures from a compiler's point of view.

5. **(5 points)** Register allocation and instruction scheduling are often performed separately (in different phases). Explain the advantages and problems of this separation.

6. Consider scheduling the code below using list scheduling. All instructions must complete before executing the `jmp` instruction. Assume the following instruction latencies:
   - 2-cycle latency for load
   - 1-cycle latency otherwise

(Continued on Next Page)

◎請用深黑色鋼筆或原子筆出題

命題老師簽名：

```
    <op> <destination, source1, source2>
1   load  r1, a
2   add   r2, r1, #4
3   store x, r2
4   load  r3, b
5   mult  r4, r3, r2
6   load  r1, c
7   add   r5, r1, r3
8   store y, r5
9   load  r6, d
10  mult  r7, r5, #1
11  store z, r7
12  jmp
```

(a) **(9 points)** Build the precedence graph for the instructions. Mark dependences as flow, anti, output, or control. You can ignore transitive dependences.

(b) **(9 points)** Schedule the instructions for a two-issue VLIW processor, using forward list scheduling. Showing candidates instructions at each cycle. Prioritize candidates using (1) critical path, (2) latency of instruction, and (3) number of children. Please use the number associated with the instruction for shorthand writing.