國立交通大學試題紙

九十八學年度第一次
博士班資格考

科目：編譯器設計(A)　　日期：99 年 1 月 28 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.

---

1. (20 points) Given a context-free grammar G. Assume that no non-terminal in G may derive the empty string. Please design an algorithm that computes the set of ADJ, which is the set of two adjacent terminals in the sentences derived from G. For example, let the language $L =_{def} \{aabcd, bbbc, aaacd\}$. Then $ADJ = \{aa, ab, bc, cd, bb, ac\}$. Specifically,

$$ADJ =_{def} \{ab | a, b \in T, \exists \alpha, \beta \in T^*, \alpha ab \beta \in L(G)\}$$

Explain your algorithm with the following grammar.

S → a X
X → b c c b d
X → b c X d b
X → b Y b X d
Y → e X Y
Y → c Y

2. (15 points) Find a nondeterministic finite automaton, deterministic finite automaton, and a minimum deterministic finite automaton for the following regular expression:

```
(ac*)+a* | a(c*a)b+
```

3. (15 points) Assume that an `if` statement is written as

```
if exp then if exp then stmt else stmt
```

We adopt the convention that the `else` part is paired with nearest unpaired `if`. For the above the example, the `else` is paired with the second `if`. Please design a LR(1) grammar for the `if` statement and show that the grammar is LR(1) by constructing its parsing table.

國立交通大學試題紙

九十八學年度第一次
博士班資格考

科目：編譯器設計(B)　　日期：99 年 1 月 28 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. [**15 points**] Suppose that the type of each identifier is a subrange of integers for expressions with arithmetic operators (addition, subtraction, multiplication, division, and modulus). For the following grammar

   $E \rightarrow E_1 + E_2$
   $E \rightarrow E_1 - E_2$
   $E \rightarrow E_1 \times E_2$
   $E \rightarrow E_1 \div E_2$
   $E \rightarrow E_1$ **mod** E_2

   Write semantic rules that compute the subrange of each subexpression (assuming that the values of subexpressions are non-negative integers).

2. [**15 points**] Multidimensional arrays can be stored in row-major order (last subscript varies fastest), as in C++, or in column-major order (first subscript varies fastest), as in Fortran. Develop the access functions for both of these arrangements for three-dimensional arrays.

   (HINT: Let the subscript ranges of the three dimensions be named `min(1)`, `min(2)`, `min(3)`, `max(1)`, `max(2)`, and `max(3)`, where `min(1)`, `min(2)`, and `min(3)` are lower bounds for dimension one, dimension two, and dimension three, respectively, and `max(1)`, `max(2)`, and `max(3)` are upper bounds for dimension one, dimension two, and dimension three, respectively. Assume the element size is `size`.)

3. [**10 points**] Describe a typical stack frame (activation record). What are the components, and what do they do?

4. [**10 points**] Consider the following two code templates for the while loop:

(a)
```
execute [while E do C] =
      JUMP h
   g: execute C
   h: evaluate E
      JUMPIF(1) g
```

(b)
```
execute [[while E do C]] =
   g: evaluate E
      JUMPIF(0) h
      execute C
      JUMP g
   h:
```

The command `evaluate E` will place the result of evaluating the condition E (with result 1 for true and 0 for false) in such a way that the following `JUMPIF` can access the result.

(1) Argue that the two code templates are semantically equivalent: Show the object codes produced by the two templates for the source code

```
while i > 0 do i := i - 2
```

(2) Compare the two object codes and explain why most compilers use (a variation of) the second template.