

# 國立交通大學試題紙

科目：編譯器設計(A)

日期：98 年 7 月 23 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) (1) When is overload resolution invoked during compilation? Please be as specific as possible.  
(2) Use an example to explain the algorithm of overload resolution.

2. (20 points) Consider the following context-free grammar:

```
E → E + T
E → E - T
E → T
T → T * F
T → T / F
T → F
F → ( E )
F → id
```

Add action routines to translate an infix expression into the postfix equivalent.

3. (15 points) (1) Devise an internal representation for common data types, including int, float, array, and record in a compiler.  
(2) Devise a translation scheme for translating type declarations into the internal representation in part (1). You may use the following grammar:

```
type → int
type → float
type → array [ int .. int ] of type
type → record fieldlist end
fieldlist → field fieldlist
fieldlist →
field → id : type ;
```

科目：編譯器設計(B)

日期：98 年 7 月 23 日 第 2 頁 共 2 頁

1. (10%) What is the difference between an interpreter and a compiler? Java can be considered both an interpreter and a compiler. How is that so? What is the advantage of designing the language that way?
2. (10%) Let the alphabet  $\sigma = \{a, b, c, d\}$ . Construct a deterministic finite state automaton with the minimal number of states to accept the regular expression  $(a|(bc)^*d)^+$ . You have to show the construction process, not just the final result.

3. (30%) Consider the following grammar  $G$  where  $S$ ,  $L$ , and  $R$  are non-terminals and  $=$ ,  $*$ , and  $id$  are terminal symbols.

$S \rightarrow L=R \mid R$

$L \rightarrow *R \mid id$

$R \rightarrow L$

- (1) Is  $G$  SLR(1)? If yes, give the parse table. Otherwise, show why.
- (2) Is  $G$  LR(1)? If yes, give the parse table. Otherwise, show why.
- (3) Is  $G$  LALR(1)? If yes, give the parse table. Otherwise, show why.