NAME

copt – peephole optimizer

SYNOPSIS

copt *file* ...

DESCRIPTION

copt is a general-purpose peephole optimizer. It reads code from its standard input and writes an improved version to its standard output. *copt* reads the named files for its optimizations, which are encoded as follows:

```
<pattern for input line 1>
<pattern for input line 2>
 ...
<pattern for input line n>
=
<pattern for output line 1>
<pattern for output line 2>
 ...
<pattern for output line m>
<blank line>
```

Pattern matching uses literal string comparison, with one exception: ‘‘%%’’ matches the ‘‘%’’ character, and ‘‘%’’ followed by a digit matches everything up to the next occurrence of the next pattern character, though all occurrences of %*n* must denote the same string. For example, the pattern ‘‘%1=%1.’’ matches exactly those strings that begin with a string X, followed by a ‘‘=’’ (the first), followed by a second occurrence of X, followed by a period. In this way, the input/output pattern

```
mov $%1,r%2
mov *r%2,r%2
=
mov %1,r%2
```

commands *copt* to replace runs like

```
mov $_a,r3
mov *r3,r3
```

with

```
mov _a,r3
```

Note that a tab or newline can terminate a %*n* variable.

copt compares each run of input patterns with the current input instruction and its predecessors. If no match is found, it advances to the next input instruction and tries again. Otherwise, it replaces the input instructions with the corresponding output patterns, pattern variables instantiated, and resumes its search with the *first* instruction of the replacement. *copt* matches input patterns in reverse order to cascade optimizations without backing up.

BUGS

Errors in optimization files are always possible.