

CSI 402 – Systems Programming – Handout 7.2

Algorithms for Constructing External Reference and Definition Tables

Note: This handout shows outlines of the algorithms used by an assembler for creating the External Reference Table (ERT) and the External Definition Table (EDT) of a module (control section).

I. Algorithm for Creating the ERT of a Module:

1. Form a list EL of all the external symbols from the EXTREF directive. (EL contains all the external symbols that can be referenced in the current module.)
2. Initialize ERT to empty.
3. **while** (there are lines in the source file) **do** {
 - (a) Get the next line from the source file.
 - (b) **if** (there is a symbol in the operand field of the line) **then**
 - (i) Let X denote the symbol in the operand field.
 - (ii) **if** (X appears in EL) **then** {

```
/* X is an external symbol. */
Insert X and the current LC value into ERT.
}
```
 - else** {

```
/* X is a local symbol. */
Find the address of X using the Symbol Table and
insert the address into the instruction.
}
```

(over)

Algorithm for Creating the EDT of a Module:

1. Form a list DL of all the external symbols from the EXTDEF directive. (DL contains all the external symbols that are defined in the current module.)
 2. Initialize EDT to empty.
 3. **while** (there are lines in the source file) **do** {
 - (a) Get the next line from the source file.
 - (b) **if** (there is a symbol in the label field of the line) **then** {
 - (i) Let X denote the symbol in the label field.
 - (ii) **if** (X appears in DL) **then** {

```
/* X is an external symbol. */
```

Insert X and the current LC value into EDT. (We must also make sure that X is not multiply defined in the EDT.)
}
 - (iii) Insert X and the current LC value into ST. (We must also make sure that X is not multiply defined in the ST.)
 - } /* End of outer if */
 - } /* End of while loop */
-