91.203 HW 5 & 6: Symbol Table II

In HW4, a symbol table is constructed with token (2 words), value (1 word), and status (1 word) fields by copying labels and variables from inArr with LOC and DEFN. HW5 and HW6 include all the features of HW4 and add the computation of the status. Instead of merely storing a label or a variable in the symTab table, you need to search the symTab to determine if the search token is already in symTab or not. According to the results, different actions need to be taken. This is now handled in VARIABLE(). The structure of HW5 and HW6 can be organized as follows.

Loop: Read pairs of (token, type) into inArr including ‘#’ // getTokens()

i=0;

if (inArr[i+1][0] != ‘:’) goto operator

DEFN = 1

valVar = VARIABLE(TOKEN, DEFN)

operator:

i++ // skip operator

DEFN = 0

isComma = true

chkVar: if (inArr[i][2] == 6) goto dump // ‘#’

if (!isComma || inArr[i][2]!=2) goto nextVar

valVar = VARIABLE(TOKEN, DEFN)

nextVar:

isComma = (inArr[i][0] == ‘,’)

i++

goto chkVar

dump: clear inArray

print symTab

LOC +=4

goto loop

VAVRIABLE:

Call LOOKUP(TOKEN, oldVal, oldStatus)

Call symSTATUS

retVal = symACTS(newStatus) (from the jump table)

return retVal

VARIABLE is passed the current TOKEN (or a pointer to the current TOKEN), and a flag DEFN. VARIABLE calls LOOKUP to search for the TOKEN in the Symbol Table.

1. If TOKEN is found from the symbol table, return the value and the status of the token presently stored in the symbol table.
2. If the search fails, LOOKUP adds a new NODE for the current TOKEN, and return -1 for both old value and old status.

SymSTATUS sets a new status for the token, and updates the status of the token in the symbol table. Since there are six status to consider, a jump table is often used to process actions corresponding to the new status value.

symACTS: b symACT0

b symACT1

b symACT2

b symACT3

b symACT4

b symACT5

The last instruction in each action routine should be a branch instruction, with the target address of returning from VARIABLE.

Each action routine will carry out what is described in the ‘action’ column of the table below. In order to call a right routine, the following MIPS instructions can be used, assuming that $s1 has the new status value (you are free to use another register other than $s0).

la $s0, symACTS

sll $s1, $s1, 2

add $s0, $s0, $s1

jr $s0

The status field uses only the last three bits: (First occurrence of the token, Already defined, DEFN).

|  |  |  |  |
| --- | --- | --- | --- |
| 4’s | 2’s | 1’s | action |
| 0 | 0 | 0 | This is a forward reference. Store LOC in the token value field of the NODE and return the previous contents of the token value field |
| 0 | 0 | 1 | A previously used TOKEN is defined for the first time. Store LOC in the token value field of the NODE and return the previous contents of the token value field. |
| 0 | 1 | 0 | A previously defined TOKEN is used. Return the contents of the token value field. |
| 0 | 1 | 1 | A previously defined TOKEN is defined again. Print “Double Definition Error” and return -1 (FFFFx). |
| 1 | 0 | 0 | TOKEN seen for the first time as a forward reference. Store LOC in the token value field of the NODE and return FFFFx. |
| 1 | 0 | 1 | TOKEN seen for the first time as a definition. Store LOC in the token value field of the NODE and mark the TOKEN as defined. |
| 1 | 1 | 0 | N/A |
| 1 | 1 | 1 | N/A |

### **INPUT and OUTPUT**

Input and output are identical to those in HW4.

**Submission:**

**HW 5: Due 10/22 (W) 11:00AM**

Write a pseudo code in MS Word for the function symStatus above, which computes the new status of a symbol, and email to [kim@cs.uml.edu](mailto:kim@cs.uml.edu) or hand in in the class.

**HW 7: Due 10/31 (F) Midnight**

Email your asm file to the TA.