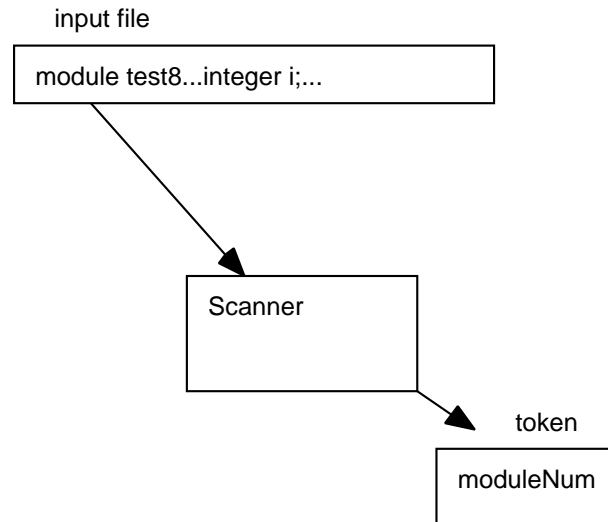

Information Flow

Moving Information Through the Compiler

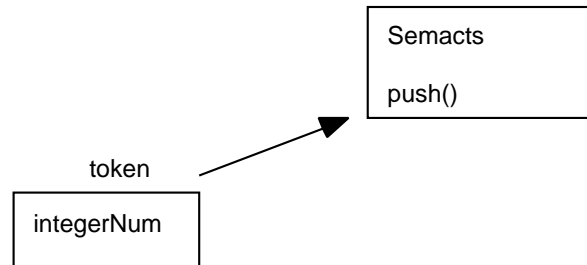
Origination of the Data

- | The scanner reads the input file and creates a TOKEN record.
- | The TOKEN describes the symbol seen in the input.



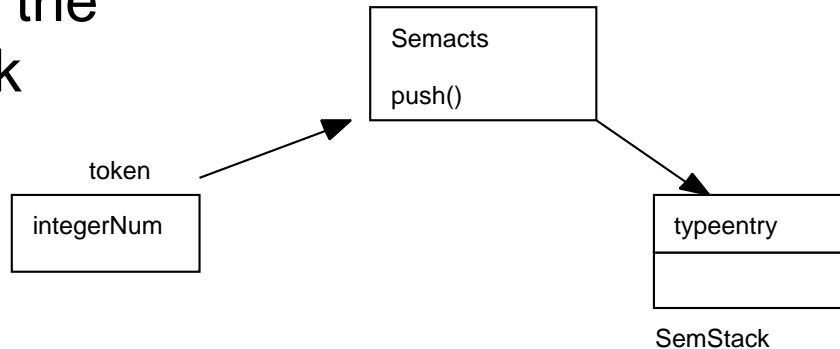
Passed to Semacts

- | The parser calls a semantic routine after scanning important tokens.
- | Usually #push is the first semantic action executed.



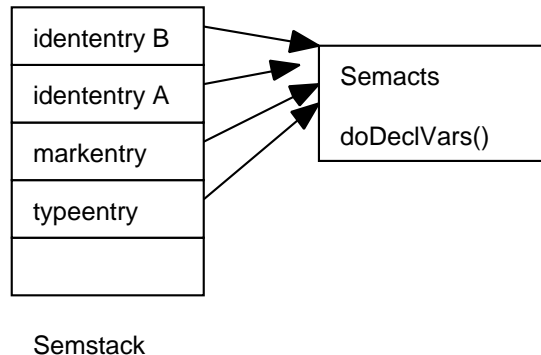
Push

- | push transforms the token record into a semantic record and pushes it onto the semantic stack



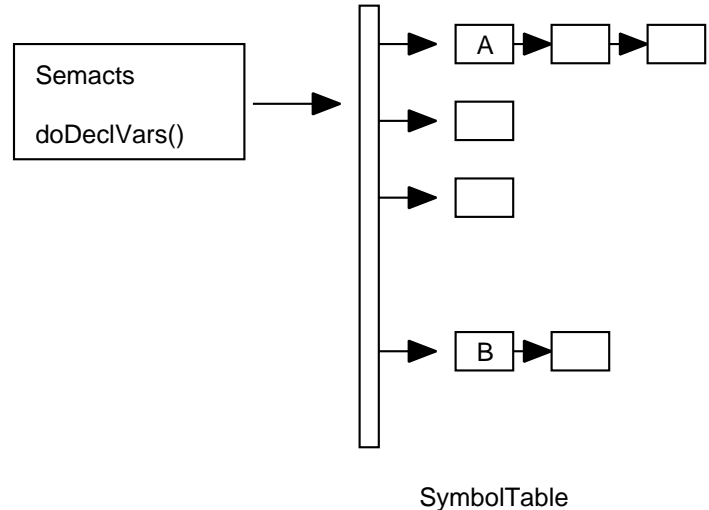
Continuing

- | Several other records are also pushed similarly.
- | Some are transformed along the way by other semantic actions
- | Then a major action is called.



Declaring Variables

- | An action like `DoDeclVars` takes a set of semantic records and creates Symboltable entries.
- | These entries may then be retrieved later.

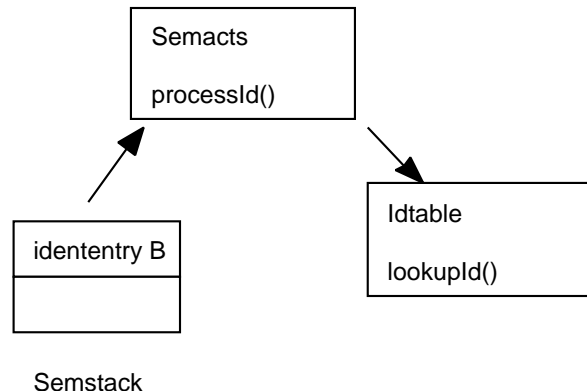


Reversing the Process

- | Once a name is in the symboltable, processId may retrieve it.
- | ProcessId rebuilds a semantic record on the stack from information in the symboltable.
- | This is a two step process.
- | First transformId looks up a name in an idententry at the top of the semantic stack.

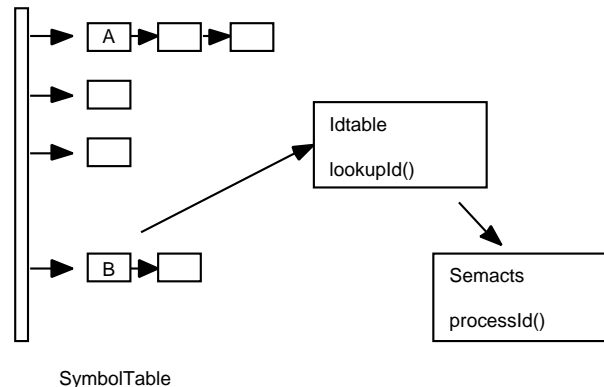
TransformId Operation

- | transformId calls lookupId from the idtable module.
- | LookupId finds the record in the symboltable and returns it to processId.



Retrieving SymbolTable Info

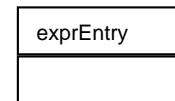
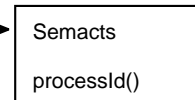
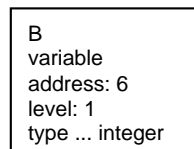
- | LookupId returns a pointer to a symboltable record
- | transformId can then examine the record to determine what kind of data it has.
- | In this case B is a variable



Examining The Record

- | transformId looks at the record it gets from lookupId.
- | transformId then creates an appropriate Semantic entry and replaces the ident with the new entry.

Symboltable record



SemStack

And On...

- | This new entry in the semantic stack will be the input to another semantic action.
- | The entry will be popped when it is no longer needed.

The Basic Flow

- | Information flows from the scanner into a token. From the token into semacts via push and other transforming actions like transformId and doDeclVars.
- | The semantic record is the basic medium of exchange in this kind of a compiler.