

Intermediate Representation (IR) grammar

(N^* denotes 0, 1 or more repetitions of N)

<i>IRProgram</i>	→ <i>StringsBlock</i> <i>IRStm</i> [*]
<i>StringsBlock</i>	→ strings { <i>StringDef</i> [*] }
	→
<i>StringDef</i>	→ <i>id</i> : <i>STRING</i>
<i>IRStm</i>	→ MOVE (TEMP <i>id</i> , <i>IRExp</i>)
	→ MOVE (MEM (<i>IRExp</i>) , <i>IRExp</i>)
	→ NOOP
	→ JUMP (<i>IRExp</i>)
	→ CJUMP (<i>IRExp</i> , <i>op</i> , <i>IRExp</i> , <i>id</i> , <i>id</i>)
	→ EXP (<i>IRExp</i>)
	→ LABEL <i>id</i>
	→ SEQ (<i>IRExp</i> , <i>IRExp</i>)
	→ PROLOGUE (<i>POSINT</i> , <i>POSINT</i>)
	→ EPILOGUE (<i>POSINT</i> , <i>POSINT</i>)
<i>IRExp</i>	→ BINOP (<i>IRExp</i> , <i>op</i> , <i>IRExp</i>)
	→ CALL (<i>IRExp</i> <i>Arg</i> [*])
	→ CONST <i>IntLit</i>
	→ MEM (<i>IRExp</i>)
	→ TEMP <i>id</i>
	→ NAME <i>id</i>
	→ ESEQ (<i>IRStm</i> , <i>IRExp</i>)
<i>IntLit</i>	→ <i>POSINT</i>
	→ <i>NEGINT</i>
<i>Arg</i>	→ , <i>IRExp</i>

Additional information about the terminal symbols:

STRING is a sequence of arbitrary characters enclosed in quotation marks, for example: "**Th&_%\$is 12is a3@ STRING.**" [Note: there is no syntax for escaping characters, so a *STRING* cannot include the " character.]

op is one of the following: **LT LE EQ ADD SUB MUL DIV**

id is a non-empty sequence of letters, underscores, @ symbols, dots (the . character) and digits, starting with a non-digit

POSINT is a non-empty sequence of digits [Note: **0** is a *POSINT*!]

NEGINT is a minus-sign followed by a non-empty sequence of digits

There is no syntax for comments.