**Functions**

**Declarations:**

FDECL TSTRING [func name]

FRETS TVOID 0V

[return values]

FERETS TVOID 0V

FPARAMS TVOID 0V

[params]

FEPARAMS TVOID 0V

FEDECL VOID 0V

**Definitions:**

FDEF TSTRING [func name]

FDEF2 TVOID 0V

[instructions go here]

FRET VOID 0V

FEDEF2 TVOID 0V

FEDEF VOID 0V

**Calls:**

FCALL TSTRING [func name]

FCPARAMS TVOID 0V

[actual parameters go here]

FECPARAMS TVOID 0V

FECALL TVOID 0V

**Extra Instructions:**

PNUM TNUMBER [number] – Alllows us to specify a parameter number of an actual parameter

PARAMF [type] [value] – Allows us to specify an actual parameter

FRET TVOID 0V – Allows us to return from a function.

PARAMF2 TVOID 0V – Allows us to put the specified parameter in %accumulator.

RNUM TNUMBER [number] – Allows us to specify a return value number.

RETURNF [type] [value] – Allows us to modify a return value to return.

FNAME TSTRING [name] – Allows us to set a function name as fname.

RETURNV TVOID 0V – Allows us to get a return value in %accumulator.

**Reg.fpointer2 instructions:**

FNCLOAD TNUMBER [num] – Loads a function from %fpointer1 to %fpointer2.

FNCSTORE TNUMBER [num] – Stores a function from %fpointer2 to %fpointer1.

FNCSTORE TVOID 0V – Adds %fpointer2 to %fpointer1.

FNCNEW TVOID 0V – Makes a new function on the C++ heap.

FNCDELETE TVOID 0V – Deletes a new function on the C++ heap.

FNCSNAME TVOID 0V – Names the function in %fpointer2 from %accumulator

FNCSRET [type] 0V – Adds a return value of type type to %fpointer2

FNCSPARAM [type] 0V – Adds a parameter of type [type] to %fpointer2 with value %accumulator

FNCSDEF TNUMBER [where] – Adds a definition beginning at where to %fpointer2.

FNCSDEF TSTRING [label] – Adds a definition beginning at label.

FNCGNAME TVOID 0V – Puts the name of %fpointer2->name() in %accumulator

FNCGRET TNUMBER [num] – Puts %fpointer2->ret(num) in %accumulator

FNCGPARAM TNUMBER [num] – Puts %fpointer2->param(num) in %accumulator

FNCGDEF TVOID 0V – Puts %fpointer2->definition() in %accumulator