

New Technical Notes

Macintosh



Developer Support

Pascal Routines Passed by Pointer Platforms & Tools

Revised by:
Written by: Scott Knaster

March 1988
July 1985

Routines passed by pointer are used in many places in conjunction with Macintosh system routines. For example, filter procedures for modal dialogs are passed by pointer, as are controls' action procedures (when calling `TrackControl`), and I/O completion routines.

If you're using MPW Pascal, the syntax is usually

```
partCode := TrackControl(theControl, startPt, @MyProc)
```

where `MyProc` is the procedure passed by pointer (using the `@` symbol).

Because of the way that MPW Pascal (and some other compilers) construct stack frames, any procedure or function passed by pointer **must not** have its declaration nested within another procedure or function. If its declaration is nested, the program will crash, probably with an illegal instruction error. The following example demonstrates this:

```
PROGRAM CertainDeath;

  PROCEDURE CallDialog;

    VAR
      x : INTEGER;

    FUNCTION MyFilter(theDialog: DialogPtr; VAR theEvent: EventRecord;
                     VAR itemHit: INTEGER): Boolean;
      {note that MyFilter's declaration is nested within CallDialog}

      BEGIN {MyFilter}
        {body of MyFilter}
      END; {MyFilter}

    BEGIN {CallDialog}
      ModalDialog(@MyFilter,itemHit) {<----- will crash here}
    END; {CallDialog}

  BEGIN {main program}
    CallDialog;
  END.
```

Further Reference:

- Macintosh Memory Management: An Introduction
-