



# POSTSCRIPT

## Programming:

Fun With Graphics

**Date:** Wednesday, September 27

**Time:** 5:00 pm

**Place:** Lawson 1142

**Speaker:** Douglas Comer, Distinguished Professor

### Abstract

The printers used with early computers only printed “typewriter” text with a fixed width per character. In the 1980s, as laser and ink jet printers emerged, it became possible to render multiple fonts, typefaces with variable-width spacing, and graphics. The question arose: how should we specify items to be printed? One answer came from the CS research community: instead of requiring a computer to generate and download a bit-mapped image for each page, build a printer that is programmable. That is, when a computer needs to print a page, have the computer generate and download a small program that specifies how to render the page. The idea of a programmable page rendering system was popularized by Adobe Systems with the PostScript programming language.

PostScript is both fascinating and unexpected. Although it is Turing-complete (i.e., can compute any computable function), the language does not have an assignment statement or a way to declare variables or functions. Instead, everything is dynamic, including types. Both fonts and graphics are integrated as first-class objects. This talk explains the basics of PostScript, and shows examples of PostScript programs along with the resulting output when the programs are run. Join us and learn about a fun programming system unlike anything you have seen before.

**Sponsors:** Undergraduate Student Board (USB) and Graduate Student Board (GSB)