Signal1 Demonstration

Files

The files for this demonstration can be found in the rlogin cluster in the directory

`/web/courses/cs3214/spring2014/butta/examples/signal-demo/signal1`

The files are `main.c, Makefile rngs.c rngs.h.`

The `Makefile` will create an executable named `quad1`. This program computes the integral (the area under the curve) of a simple function using a Monte Carlo random sampling technique.

Purpose

The purposes of this demonstration are

- to explore the effect on an executing process of sending different signals from the terminal,
- to see how gdb can be used to help gain information about programs that receive signals

Part 1: Steps

1. Run the makefile to create the executable program `quad1`. Execute `quad1` under `gdb` using the command "`gdb quad1`". You should see some copyright information printed by gdb followed by the (gdb) prompt.
2. At the (gdb) prompt begin the execution of `quad1` by using gdb's `run` command. You should see a message that begins "`Starting program…`"
3. When the `quad1` program is running send the "stop" signal to the program by entering a cntl-z (simultaneously pressing the "control" and "z" keys). You should see output from gdb that begins "`Program received signal SIGTSTP, Stopped (user)…`"
4. The output from gdb shows the line of C code that was being executed at the time the signal arrived, the line number (as filename:linenumber) of the line of C code that was executing. You can see the current program counter using the gdb command "`print $pc`". You can see how many random samples so far are above the curve by using the gdb command "`print above`" where `above` is a variable declared in `main.c`. You can see how many random samples so far are below the curve by using the gdb command "`print below`". Write the program counter value and the values of `above` and `below` in the table below.
5. Continue the execution of the process executing the `quad` program by the gdb command "`continue`". Note: If gdb again immediately stops with a SIGTSP signal, enter the "`continue`" command again. Note that you can use the gdb command "`quit`" at the gdb prompt to end the execution of gdb and the process executing the `quad1` program.
6. Repeat steps 3-5 a number of times and continue filling in the table below.

Program Counter	above	below

Questions

Based on your observations, answer these questions.

1. Is there any way to predict at what point in the execution of a program a signal will be received?

2. Does the signal have any effect on what is computed by the program?