Benjamin Berger

Project 7

Goal:

Make edits to the kernel infrastructure so that we can count page faults.

Outline:

Edit where page fault handling occurs and make an addition so that we can count how many page faults occur in our code.

Solution:

First I initialized the pid and global counter that would be used in the fault.c file. After that I added the system call handler to the bottom of the fault.c file. When a 0 was sent in it would start counting and when a 1 was sent in it would stop counting and return the number while zeroing the counter.

After that I when into the do\_exception part of fault.c and added and if statement that would increment the global counter variable if the set pid was the same as the current pid.

Once done with fault.c, I made edits to the syscall.s file, and various unistd.h files. In the unistd.h files I added the #define myFunctionName 310 to let them know what I would be accessing. In syscall.S I added the system call for my function that would let me into fault.c with my program.

Once that was done I created two difference test programs named test1 and test2. They initialize 3 vectors in two different ways and computes the multiplication of the first and second and stores it in the third.

Right before the calculations are made, I make a syscall to my function to begin the counting. After the calculations I make a syscall to my function to get the result.

Flow:

Into the syscall that runs in falut.c, and back out to my program.