

TDIU16: Process- och operativsystemprogrammering

Systemanrop för tangentbord och skärm (read och write)

Klas Arvidsson, Daniel Thorén, Filip Strömbäck

1 Mål

I och med att alla program i Pintos kör i user-mode så har de i nuläget inte möjlighet att göra något intressant. Tack vare `halt` och `exit` i förra laborationen kan de i alla fall stänga av sig, men det är inte särskilt roligt. I den här laborationen ska vi göra det möjligt för program att läsa indata från tangentbordet och att skriva utdata till skärmen.

2 Bakgrund

I UNIX-baserade system används systemanropen `read` och `write` för att läsa från och skriva till filer som man tidigare har öppnat. Eftersom allt i UNIX är en fil, hanteras även in och utmatning till terminalen som en fil. Som standard är standard in och standard out-filerna redan öppna som fildeskriptor `STDIN_FILENO` och `STDOUT_FILENO`. Detta beteendet vill vi efterlikna i Pintos.

För mer information om hur dessa systemanrop fungerar, se Pintos-Wiki under rubriken "Systemanrop" under "Filsystem".

3 Testprogram

Filen `examples/line_echo.c` innehåller ett lämpligt testprogram. Se kommentaren i början av filen, samt i Pintos-Wiki under "Att köra Pintos" för detaljer om hur programmet körs.

Filen `examples/file_syscall_tests` innehåller fler tester. De första testerna gäller denna uppgift, och de senare för kommande uppgifter. Notera att `file_syscall_tests` är ett för långt filnamn för Pintos. Se kommentar i filen för hur du kan köra programmet.

4 Uppgift

Implementera systemanropen `read` och `write` för läsning från tangentbordet respektive skrivning till skärm. *Filer hanteras senare*. Observera att tangentbordet och skärmen är resurser som delas av alla trådar och processer. Fundera på hur det fungerar om två olika processer vill använda tangentbordet samtidigt.

Det finns två konstanter i Pintos som defenierar vilka fildeskriptorer som representerar tangentbordet respektive skärmen. Skärmens fildeskriptor defenieras av `fd == STDOUT_FILENO` och tangentbordets fildeskriptor defenieras av `fd == STDIN_FILENO`. I och med att vi ännu inte hanterar filer räcker det att systemanropen hanterar dessa två fallen korrekt.

Testa dina systemanrop med testprogrammet `line_echo` som läser rader från tangentbordet och skriver ut dem igen. *Var noga med att skriva indata i terminalen*, inte i det emulerade fönstret (det emulerade fönstret har inte rätt tangentbordslayout). Utdata skall visas i **bägge** fönstren. Naturligtvis vill användaren kunna se de tecken som matas in på tangentbordet direkt på skärmen under tiden denne skriver.