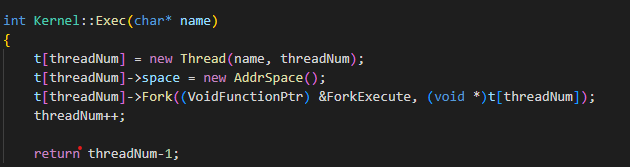
有兩個register set一個for user 一個for kernel

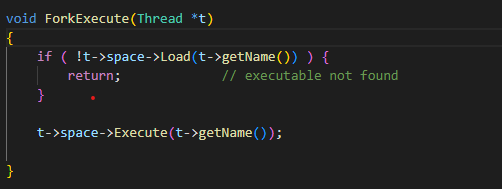
Kernel.cc的Exec()



New一個thrad,該thread的address space並且執行Fork跟ForkExecute

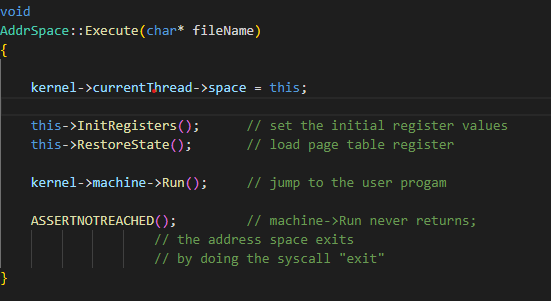
Kernel.cc的ForkExecute(),Load():

建造一條Thread,並且建立address space for user program



在User/History/qsa1.cc,使用Load()把user program從file載入到memory,Execute()表示如果program已經被load進memory就執行這個函式

Addrspace.cc的Execute():



Load進memory後要使用當前的Thread執行user program,初始化user register,如果要context switch必須保存好這個register的資訊(page table所在的位置),最後執行MIPS-RUN程式

Thread.cc的Fork():

同時執行caller跟callee,會先分配Stack,當scheduler切到某thread時可以確保執行該thread的程式碼,並且把此thread放進ready queue

Thread.cc StackAllocate():

分配以及宣告stack

Scheduler.cc的ReadToRun():

設定thread為ready state,並且放進ready list.

threads/kernel.cc

ExecAll就是要Main Thread(Kernel)依序去執行(Exec)所有要執行的程式(Thread)

Scheduler.cc Run():

必須先確保interrupt關掉,如果目前thread結束了,把現在的thread mark起來並刪除,如果這個thread是user program會儲存user register,下一個thread進來,並且設為Running state