○ How Nachos allocates the memory space for new thread(process)?

在Fork的時候，會new一個AddrSpace給thread->space。

○ How Nachos initializes the memory content of a thread(process), including

loading the user binary code in the memory?

使用AddrSpace::Load()，將讀取的file切進記憶體。

○ How Nachos creates and manages the page table?

實作之前是直接在建構式宣告一個NumPhysPages大小的page table。

實作之後則是把建立的這件事，等到Load的時候，再依據讀取進來的程式大小，宣告適當尺寸的page table。

管理的部分，則是Load在宣告完page table之後，會接續進行。

○ How Nachos translates address?

translate.cc中的machine::Translate

\*physAddr = pageFrame \* PageSize + offset;

AddrSpace.cc中 (在實作後)

Virtual page number = virtualAddr / PageSize

Page內的Offset = virtualAddr%PageSize

○ How Nachos initializes the machine status (registers, etc) before running a

thread(process)

在Thread::StackAllocate() 的時候，會把自己的ThreadRoot() 與相關資訊存進Stack裡面。相關資訊包含：ThreadBegin、ThreadFinish、該thread的ForkExecute函式指標，與指向自己的指標 (Thread \*)。

○ Which object in Nachos acts the role of process control block

Thread (這一個class)。因為它包含了page table、register等資訊。

○ When and how does a thread get added into the ReadyToRun queue of Nachos

CPU scheduler?

在Thread::Fork() 裡面，會呼叫Scheduler::ReadyToRun()，並在此時把這個thread 放進readyList裡面。