MP2 Spec & MP1 discussion

Outline

- MP2 Spec
- MP1 code solution
- MP1 demo
- VScode ssh login

MP2 Spec

Open

- Must maintain OpenFileTable and use the table entry number of OpenFileTable as the OpenFileId
- Must handle invalid file open requests, including the non-existent file, exceeding opened file limit (at most 20 files), etc.
- DO NOT use any IO functions from standard libraries (E.g. printf(), cout, fopen(), fwrite(), write()...)

```
OpenFileId OpenAFile(char *name) {
  int fileDescriptor = OpenForReadWrite(name, FALSE);
  if (fileDescriptor < 0) return -1;
  int id = -1;
  for (int i = 0; i < 20; i++) {
    if (OpenFileTable[i] == NULL) {
      OpenFileTable[i] = new OpenFile(fileDescriptor);
      id = i;
      break;
    }
  }
  return id;
}
```

Read & Write

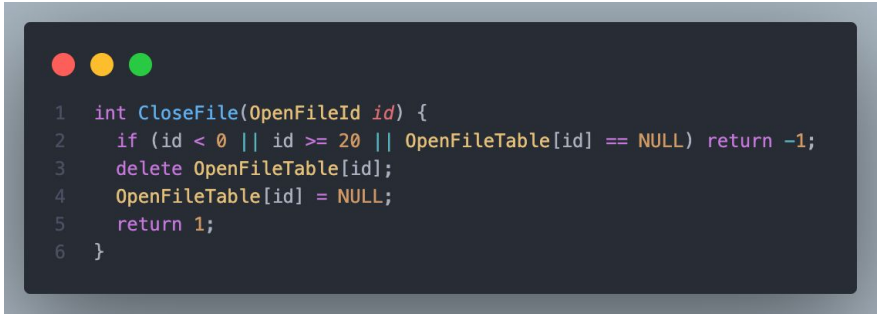
- Must handle invalid file open requests, including the non-existent file, exceeding opened file limit (at most 20 files), etc.
- Must handle invalid file read, write, close requests, including invalid id, etc.

```
int WriteFile(char *buffer, int size, OpenFileId id) {
  if (id < 0 || id >= 20 || OpenFileTable[id] == NULL) return -1;
  int ret = OpenFileTable[id]->Write(buffer, size);
  return (ret < 0) ? -1 : ret;
}

int ReadFile(char *buffer, int size, OpenFileId id) {
  if (id < 0 || id >= 20 || OpenFileTable[id] == NULL) return -1;
  int ret = OpenFileTable[id]->Read(buffer, size);
  return (ret < 0) ? -1 : ret;
}
```

Close

- Need to delete the OpenFile after you close the file (Can’t only set the table content to NULL)



```
int CloseFile(OpenFileId id) {
  if (id < 0 || id >= 20 || OpenFileTable[id] == NULL) return -1;
  delete OpenFileTable[id];
  OpenFileTable[id] = NULL;
  return 1;
}
```

MP1 demo - 1

- 剛進入 ExceptionHandler(), 這時候 Nachos 是在 kernel mode 還是 user mode ?
  Kernel mode
- 剛進入 RaiseException(), 這時候 Nachos 是在 kernel mode 還是 user mode ?
  User mode

```
1  kernel->interrupt->setStatus(SystemMode);
2  ExceptionHandler(which);        // interrupts are enabled at this point
3  kernel->interrupt->setStatus(UserMode);
```

ExceptionHandler()

```
1  kernel->interrupt->setStatus(UserMode);
2  for (;;) {
3    DEBUG(dbgTraCode, "In Machine::Run(), into OneInstruction "
4                      << "== Tick " << kernel->stats->totalTicks << " ==");
5    OneInstruction(instr);
6    DEBUG(dbgTraCode, "In Machine::Run(), return from OneInstruction  "
7                      << "== Tick " << kernel->stats->totalTicks << " ==");
8
9    DEBUG(dbgTraCode, "In Machine::Run(), into OneTick "
10                     << "== Tick " << kernel->stats->totalTicks << " ==");
11   kernel->interrupt->OneTick();
12   DEBUG(dbgTraCode, "In Machine::Run(), return from OneTick "
13                     << "== Tick " << kernel->stats->totalTicks << " ==");
14   if (singleStep && (runUntilTime <= kernel->stats->totalTicks))
15     Debugger();
16 }
```

RaiseException()

MP1 demo - 2

- CheckIfDue() 傳入的參數若設為 true，則會發生什麼事？
  - **由於 pending queue 中沒有 threads，因此快進系統時間去執行下一個 interrupt**
- 在很多地方都可以看到 ASSERT(...)，像是 ConsoleOutput::PutChar() 中，有 ASSERT(putBusy == False)
  - 用意為何？**為了防止會導致錯誤的情況發生，因此會用 ASSERT() 來確保 runtime 時有些條件一定要成立**
  - 為什麼沒有滿足條件會終止？**Abort()**

```
if (next->when > stats->totalTicks) {
  if (!advanceClock) {  // not time yet
    return FALSE;
  } else {  // advance the clock to next interrupt
    stats->idleTicks += (next->when - stats->totalTicks);
    stats->totalTicks = next->when;
    // UDelay(1000L); // rcgood - to stop nachos from spinning.
  }
}
```

CheckIfDue()

```
#define ASSERT(condition)                                          \
    if (condition) {} else {                                       \
    cerr << "Assertion failed: line " << __LINE__ << " file " << __FILE__ << "\n";      \
        Abort();                                                                        \
    }
```

ASSERT()

VScode ssh login without passwd

- `ssh-keygen -t ed25519`
- Generated keys are located at `~/.ssh/id_ed25519.pub`
- Copy the public key in `~/.ssh/id_ed25519.pub` to `~/.ssh/authorized_keys` in remote server (If the file does not exist, you need to create an empty first)