🐱 GitHub Repository 🐱

https://github.com/WWU-CS-Idriss/blitz-os-assignments-nimkicodes/tree/a4

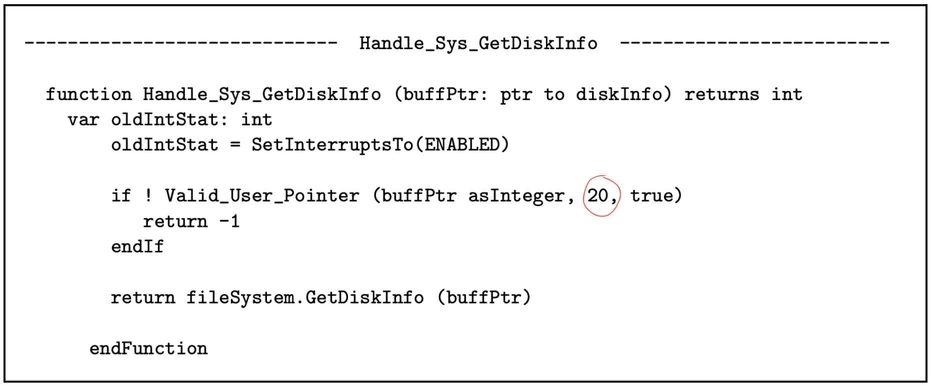
🔗link to branch a4 where assignment4 has been implemented 🔗

🐶Notes 🐶

- **Command Line Arguments:**
  I could implement command line arguments making major changes in my CopyArgument.While I understood that the command line printing should have been done from the TestProgram2.k file, in my code I did the printing inside CopyArgument.

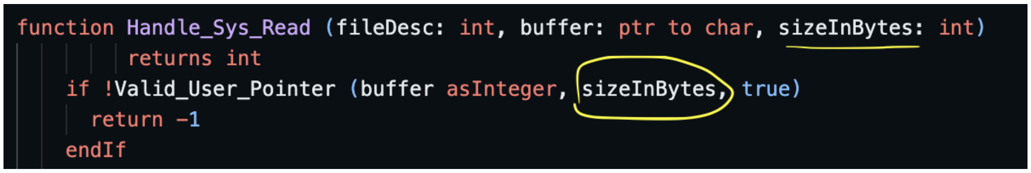
- **Extra Syscall Handlers:**
  I implemented Chmod, Link, Unlink, Mkdir, Rmdir syscall handlers in Kernel.k. The pattern of Chdir and all these five was fairly similar.

Handout_A4_509.pdf)

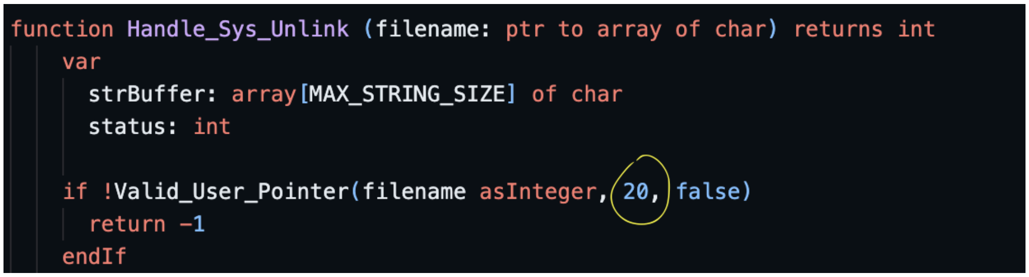


```
-----------------------------  Handle_Sys_GetDiskInfo  -------------------------

  function Handle_Sys_GetDiskInfo (buffPtr: ptr to diskInfo) returns int
    var oldIntStat: int
        oldIntStat = SetInterruptsTo(ENABLED)

        if ! Valid_User_Pointer (buffPtr asInteger, 20, true)
           return -1
        endIf

        return fileSystem.GetDiskInfo (buffPtr)

      endFunction
```

For syscalls where sizeInBytes is an argument, I passed sizeInBytes in Valid_User_Pointer and for the rest of them, I have passed the integer value of 20.



```
function Handle_Sys_Read (fileDesc: int, buffer: ptr to char, sizeInBytes: int)
          returns int
    if !Valid_User_Pointer (buffer asInteger, sizeInBytes, true)
      return -1
    endIf
```



```
function Handle_Sys_Unlink (filename: ptr to array of char) returns int
    var
      strBuffer: array[MAX_STRING_SIZE] of char
      status: int

    if !Valid_User_Pointer(filename asInteger, 20, false)
      return -1
    endIf
```

So, in my current implementation, I have kept the size value as 20.