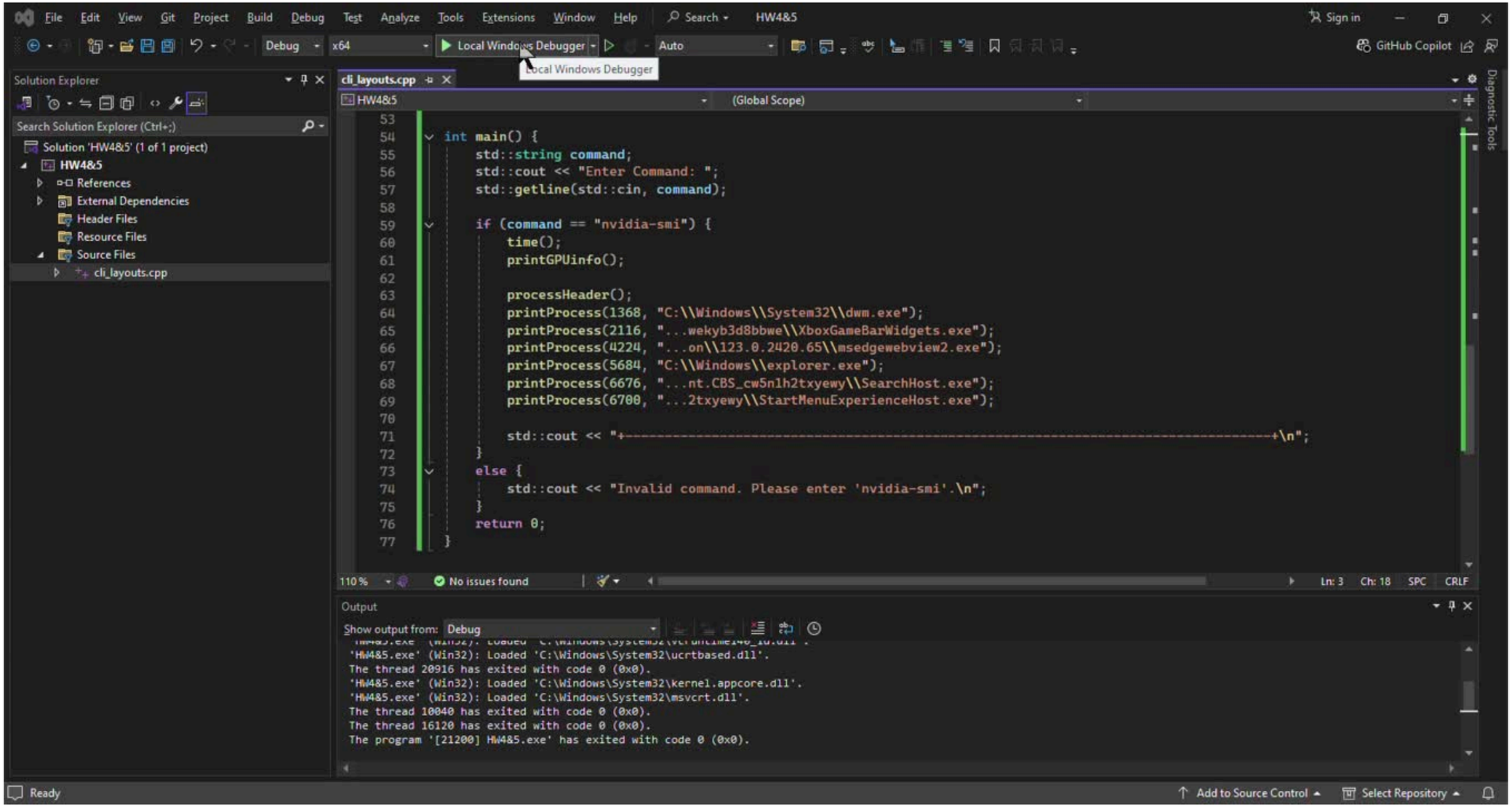
Week 4 – Homework

Creating customized CLI layouts

Kai Hiori J. Padilla

video demo

- GOOGLE DRIVE LINK



printGPUinfo

- the function's main purpose is to display the GPU info in a specific format and copy the output of the NVIDIA System Management Interface (NVIDIA-SMI) tool.

```
void printGPUinfo() {
    std::cout << "+-----------------------------------------------------------------------------+\n";
    std::cout << "| NVIDIA-SMI 551.86       Driver Version: 551.86        CUDA Version: 12.4    |\n";
    std::cout << "+-----------------------------------------+------------------+----------------------+\n";
    std::cout << "| GPU  Name                     TCC/WDDM |Bus-Id        Disp.A|   Volatile Uncorr. ECC |\n";
    std::cout << "| Fan  Temp  Perf      Pwr:Usage/Cap |       Memory-Usage |GPU-Util     Compute M. |\n";
    std::cout << "|                                       |                    |                  MIG M. |\n";
    std::cout << "+=========================================+====================+======================+\n";
    std::cout << "| 0    NVIDIA GeForce GTX 1080  WDDM |00000000:26:00.0  On|                  N/A |\n";
    std::cout << "| 28%   37C   P8          11W / 180W |   701MiB / 8192MiB |         0%     Default |\n";
    std::cout << "|                                       |                    |                  N/A |\n";
    std::cout << "+-----------------------------------------------------------------------------+\n";
}
```

processHeader

- This functions prints out a header for the processes

```
void processHeader() {
    std::cout << "+-----------------------------------------------------------------------------+\n";
    std::cout << "| Processes:                                                       GPU Memory|\n";
    std::cout << "| GPU   GI   CI   PID   Type   Process name                         Usage     |\n";
    std::cout << "+-----------------------------------------------------------------------------+\n";
}
```

printProcess

- This function's main purpose is to format and print the details of a process, ensuring that the output are all aligned and structured properly

```
void printProcess(int pid, std::string process_name) {
    const int maxLineWidth = 79;
    const int fixedPartLength = 29;
    size_t availableSpace = maxLineWidth - fixedPartLength - std::to_string(pid).length();


    std::string formattedName = process_name;

    if (formattedName.length() > availableSpace) {
        formattedName = "..." + formattedName.substr(formattedName.length() - (availableSpace - 3), availableSpace - 3);
    }
    else {
        formattedName += std::string(availableSpace - formattedName.length(), ' ');
    }

    std::cout << "| 0    N/A N/A " << std::setw(4) << pid << "  C+G    " << formattedName << "    N/A  |\n";
}
```

time

- This function just retrieves the current local time

```
void time() {
    std::time_t t = std::time(nullptr);
    std::tm local_time;
    localtime_s(&local_time, &t);

    std::cout << std::put_time(&local_time, "%a %b %d %H:%M:%S %Y") << "\n";
}
```

End of slides