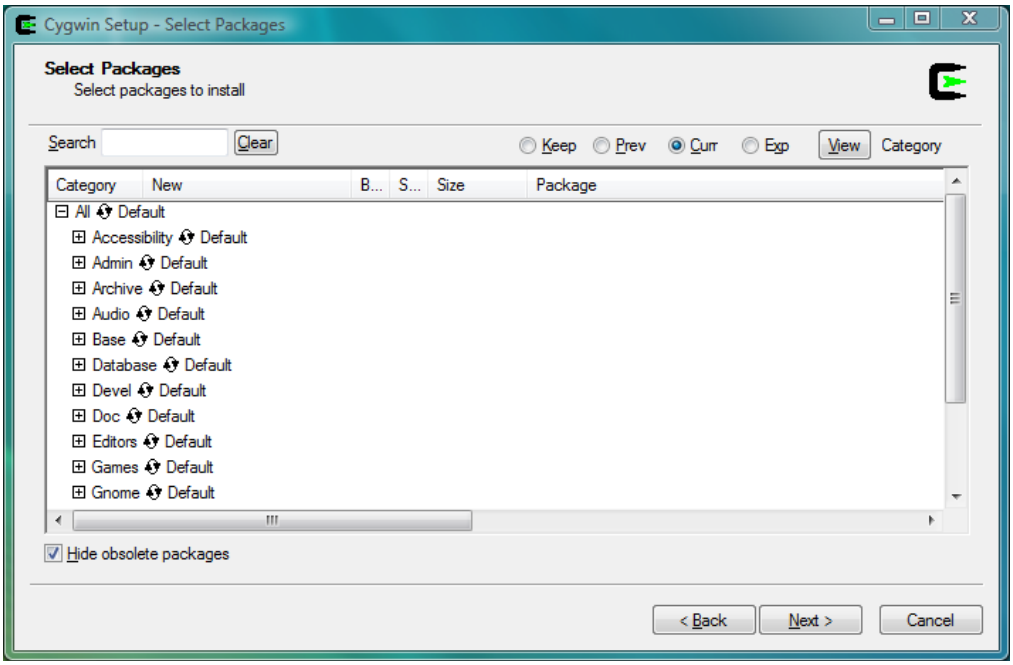
(/YoshuaNava)
**Update Instructions.md (/YoshuaNava/DD2380/commit/12b624b6d9527df14998d2cbcddeeb059e68844f)**
Yoshua Nava (/YoshuaNava) committed 6 days ago

**Instructions.md** 1.76 KB

# Instructions for running A1 files on Windows 8.1/10

## Environment setup

**1)** Download cygwin from this link (http://www.cygwin.com/) and install it. When asked about which packages you want to install, pick `gcc-core`, `g++` and `lynx` (use the search bar)




(http://www.aspiringcraftsman.com/wp-content/uploads/2010/04/cygwin-packages-1.png)

**2)** Run cygwin, and run the following commands to update gcc and g++:

```
lynx -source rawgit.com/transcode-open/apt-cyg/master/apt-cyg
install apt-cyg /bin
apt-cyg update gcc
apt-cyg update g++
```

**3)** Open cygwin, and navigate to the directory in which you have the repo and corresponding VS solution:

**Kevin**

```
cd /cygdrive/c/...
```

**Yoshua**

```
cd /cygdrive/c/Users/Yoshua/Documents/GitHub/DD2380/A1/A1_HMM
```

**4)** Then, go to the corresponding directory of the sub-task you want to run:

- DuckHunt
- HMM0
- HMM1
- HMM2
- HMM4

**5)** Create a set of FIFO special files for communication between the server and client:

```
mkfifo player2server server2players
```

---

## Running the game

**1)** Go to the corresponding directory of the sub-task you want to run. For example:

```
cd /cygdrive/c/Users/Yoshua/Documents/GitHub/DD2380/A1/A1_HMM
```

**2)** Build the game files by running:

```
g++ -o output -std=c++11 *pp
```

**3)** Run the game server in one terminal:

```
./output.exe verbose server < player2server > server2player
```

**4)** Run the game client in another terminal:

```
./output.exe verbose > player2server < server2player
```

**Note:** Alternatively, you can run both the game client and server on the same terminal, by running the command:

```
./output.exe server < player2server | ./output.exe verbose >
```