**Matteo Merlo**

**ICARUS\_POLITO**

**API weather esterne Xplane**

Le API weather sono esterne a Xplane. Creando un URL apposito, immettendo le coordinate volute, restituisce un json meteorologico completo, peró non si hanno info sugli strati di nuvole.

API usate: **openWeatherMap.org**

Creato programma funzionante in Java, solo per semplice implementazione.

Per importare libreria in Eclipse: tasto destro sul nome del progetto -> build path -> add external libraries -> selezionare tutti i file JAR

CPP:

Prima di iniziare assicurarsi che sia installato il pacchetto inglese di Visual studio. Scaricare in alternativa da <https://visualstudio.microsoft.com/it/downloads/> : scaricare installer VS 2019, selezionare Language pack e selezionare lingua inglese.

Installazione libreria **/curl.h** e **/json.h** e configurazione:

* installare git library, scaricando da [*http://git-scm.com/*](http://git-scm.com/) l’installer.

**Esegui come amministratore** la ***“powershell windows***” (non prompt comandi!!) ed entrare dentro la cartella dove si vuole installare Vcpkg.

* Inserisci comando “git clone https://github.com/Microsoft/vcpkg.git”
* Entrare dentro la cartella dove si è installato, eseguire:
* “ .\bootstrap-vcpkg.bat “
* “ .\vcpkg integrate install”
* Per cercare una libreria “ .\vcpkg search nomeLibreria” (senza .h)
* Per installare libreria “ .\vcpkg install nomeLibreria (senza .h)
* Per installare curl.h - “ .\vcpkg install curl ”
* Per installare json.h - “ .\vcpkg install nlohmann-json ” (<https://github.com/nlohmann/json>)
* Per installare jsmn.h - “ .\vcpkg install jsmn ” (<https://github.com/zserge/jsmn>)

Nel codice ho preferito personalizzare la libreria come jsmnImplemented.h. Sono tutte librerie json queste.

Per maggiori info sulla configurazione: (<https://github.com/Microsoft/vcpkg>) e (<https://www.youtube.com/watch?v=wRnjahwxZ8A>)

NOTA: per vedere formattato correttamente un json su google chrome, installare “json formatter” da Chrome Store come estensione. Utile per vedere correttamente la stringa del JSON.

NOTA: attualmente velocita campionamento 5 campioni/sec.

Da implementare:

* Implementazione per strati delle nuvole: <https://openweathermap.org/api/weathermaps>
* Implementare su thread il campionamento.

**A star algorithm -> Dijkstra Algorithm**

Ho scelto come algoritmo dijkstra, scelta ottimale per grafo con pesi. Codice finito e funzionante in un'unica classe.

ATTENZIONE: Dimensione matrice 16000 int (interi). Massima allocazione su windows 4GB, oltre modificare parametri SO.