

# Programmazione di sistema

Anno accademico 2014-2015

## Esercitazione 3

Si scriva un'applicazione C++ che implementi il comando GNU/Linux *fgrep*, utilizzato per cercare una stringa di testo costante all'interno di un file, senza considerare le espressioni regolari. Al fine di rendere più efficiente la ricerca della stringa e sfruttare al meglio le risorse di elaborazione, è necessario affidare ad un thread separato l'analisi del file, avendo cura di evitare di creare più di 20 thread contemporaneamente.

In particolare, l'applicazione dovrà ricercare all'interno dei file della cartella di origine (passata come parametro) e delle sue sottocartelle, una stringa di testo passata come parametro. Se la stringa è presente all'interno di uno o più file, l'applicazione dovrà generare un file di output, ordinato alfabeticamente, contenente i nomi dei file con path assoluto (che contengono la stringa), uno per riga, ad esempio:

```
c:\temp\file1.txt
c:\temp\subfolder1\file2.txt
c:\temp\subfolder2\file3.txt
```

La stringa da ricercare potrà contenere spazi, caratteri di punteggiatura, caratteri maiuscoli e minuscoli che dovranno essere considerati diversi.

L'applicazione dovrà ricevere come parametri da riga di comando:

- la cartella di origine dove cercare la stringa
- la stringa da ricercare
- il file di output

nel seguente formato:

```
fgrep_win.exe "cartella_origine" "stringa da cercare" file_output
```

Esempio:

```
fgrep_win.exe "c:\temp" "programmazione di sistema" out.txt
```

Per accedere all'elenco dei file contenuti in una cartella si utilizzino le funzioni `FindFirstFile` e `FindNextFile`, mentre per attendere la terminazione di uno o più thread si utilizzi la funzione `WaitForMultipleObjects`, tutte definite nel file `windows.h`. Si abbia cura di rilasciare tutte le risorse acquisite.

## Competenze da acquisire

- Utilizzo delle funzionalità di threading native Win32
- Sincronizzazione tra thread
- Accesso al file system