Exam Assignments V05

chengchengguo 183090

1. What is **CMake**?

> CMake is an open-source, **cross-platform** family of **tools** designed to **build, test and package** software. CMake is used to control the software compilation process using simple platform and compiler independent configuration files, and generate native **makefiles** and workspaces that can be used in the compiler environment your choice.[1]

- CMake is a **cross-platform build file generator**
- with CMake we can generate **build files** for **many different IDEs** and **build tools** (Microsoft Visual Studio, Xcode, **make**, . . .)
- CMake is a **command based language**, with one command per line
- **commands don't return values**: no nesting
- CMake offers also c**ontrol flow constructs** like if, elseif, else, foreach, while, break, ...

2. What role do **targets** play in CMake?

if we have more than one file with dependencies, we need to be able to tell CMake about the structure of our project, and targets will help us build it.

targets are **executables** and **libraries**

3. How would **you proceed** to **optimize code**?

1) **solve** the **problem** first without optimizations
2) **make sure** that the **program is working correctly** (unit tests)
   write test file (catch)
3) **if: performance** of the implementation is **good enough** for needs, it is **done**
4) **else: optimize** the implementation (find **bottlenecks**, improve the algorithm, parallelize, vectorize, . . .)
5) **goto:** step 2)

[1] CMake