Az `analysis` könyvtár alatt található program a vezérlési szerkezetek feltérképezésének oktatásához készült. Az algoritmus működésének alapötlete a fóliákon megtalálható, de részletes ismertetésére az óra keretein belül nem volt elegendő idő. A gyakorlat célja, hogy az érdeklődő hallgató otthon elmélyíthesse tudását, valamint, ha egy részt nem értett, akkor azt akár debugger segítségével nyomkövetve tanulmányozhassa. A program beégetett vezérlési folyamatgráfra működik, amit a StructuralAnalysis.cpp fájban található $ir :: ControlFlowGraph* makeCfg()$ függvény állít elő.

A program a GnuOcelot project része, ami a `http://code.google.com/p/gpuocelot/` linken érhető el. A forrásfájlok módosítva lettek, hogy azokat önállóan le lehessen fordítani. A szerzők felhasználták a legújabb (c++0x) C++ szabványt, ami esetleg a fordításnál gondot okozhat. GCC esetén a $-$std=c++0x kapcsolót kell megadni, Visual Studio 2008-ban az új STL tárolók a $std :: tr$ névtérben találhatók.

Microsoft Visual Studio esetén az analysis.sln, CodeBlocks használatánál az analysis.cbp project fájlok segítenek a fordításban. A három darab cpp fájl consol ablakban, vagy a linux shell-ben parancssorosan is egyszerűen lefordítható.

A program továbbvihető házi feladatnak is.

A készítendő program működjön a fordító által generált tetszőleges *.obj tárgykódra. A tárgykód disassemblálásával (például a dumpbin program segítségével), egy olyan assembly programhoz jutunk, amelyben egyszerűen felismerhetőek a vezérlés átadó (ugró) utasítások, és ennek segítségével a programot alapblokkokra lehet bontani. (Példaképp szerepelnek a sort.c és az abból készített `sort.asm` fájlok.) Így lehetőség kínálkozik a vezérlési folyamatgráf felépítésére. Az alapblokkok hordozzák az általuk reprezentált utasításokat. A forrás az igényeknek megfelelően tetszőlegesen átírhatók.

Megoldandó feladatok:

- asm fájl értelmezése;
- $std :: list < Instruction* > instr$ adatszerkezet feltöltése;
- az $instr$ alapján az alapblokkok határainak a meghatározása;
- alapblokkok létrehozása, és az utasításlista hozzáadása;
- NaturalLoop esetén **while** vagy **do** {...} **while**() ciklusok megkülönböztetése;
- az eredmény megjelenítése megfelelő kód tördeléssel (*indentálással*).