Извештај о оптимизацији предложеног

кода задатка број 3 пројекта из

предмета АОР2

Процесор: AMD Ryzen 7 4700U

Опис предложеног кода: Предложени код извучен је из једне од понуђених библиотека из текста пројекта (*C-Plus-Plus master library*). Из ове библиотеке извучен је следећи .cpp фајл:

*gram\_schmidt.cpp*

Опис кода: У овом .cpp фајлу имплементиран је Грам-Шмитов алгоритам ортогонализације (линеарна алгебра). Алгоритам је изабран јер се ради са матрицама које могу да буду великих димензија. Матрице уствари представљају низ вектора над којима се даље врше одређене операције. Подаци су double типа. Детаљнији опис алгоритма није битан за даљу оптимизацију.

Предлог за оптимизацију: У оптимизованом коду је употребљен loop-fusion (један пролаз кроз матрицу уместо 2). Извршене су 2 одвојене оптимизације. Једна је користила prefetching (раније довлачење у кеш) и loop-unrolling (уместо петље инструкције су секвенцијални једна испод друге навођене), док су за другу оптимизацију коришћене векторске инструкције које процесор подржава (пронађено на интернету).

Text

Description automatically generated Успешност оптимизација: Први предлог оптимизације довео је до успорења за неких 10% (у најбољем случају). Варирани су и параметри функције prefetch који говоре где тачно податак треба да се довуче (у који ниво кеш меморије). Постоји могућност да дати процесор не подржава инструкцију \_mm\_prefetch (да се подаци не довуку при овој инструкцији у кеш меморију). Иако је довео до успорења за велике матрице, први предлог је био бржи од оригинала у раду са матрицама малих димензија. Вођено је рачуна о томе да се подаци у кешу не преклопе тако што су се променљиве у петљама (итератори петље) стављали у регистре. Други предлог је зато довео до убрзања за 10% у односу на оригиналан код. Проверено је које векторске инструкције процесор подржава. Покренута су 3 теста. Прва два теста раде са малим матрицама и они служе запроверу исправности оптимизованог кода (провера алгоритма и рачуна). Резултате прва два теста можете видети на слици 1.

Последњи, трећи тест, тестирао је алгоритме са великим матрицама

Слика 1 – Резултати прва два теста за оба предлога

(30 x 30). Мерено је време трајања извршавања целокупног алгоритма за сва три случаја. Резултате теста 3 за сваки од случаја можете видети на сликама 2.1, 2.2 и 2.3.



Слика 2.2 – време извршавања опимизованог кода кода коришћењем векторских инструкција

Слика 2.2 – време извршавања опимизованог кода кода коришћењем prefetch инструкција

Слика 2.1- време извршавања оригиналног кода користећи матрице 30 x 30