**CS 521 ML & Compilers Spring 2025 – MP1**

**Student Name: Charlie Jyu**

**NetID: cbjyu2@illinois.edu**

**Part 1 CPU**

1.2) **Ablation Study:**

A graph of different colored bars

AI-generated content may be incorrect.

***Insight:*** As the matrix size increases, the optimizations become more effective and their effects are more pronounced. This is consistent across every optimization. As expected, more advanced optimizations generally are faster than naïve methods.

1.3) **Scaling Study:**

A graph with different colored lines

AI-generated content may be incorrect.

***Insight:*** Larger matrices require more FLOPs to compute and the runtime scales in a cubic manner as matrix dimensions increase. Additionally, the more advanced optimizations generally outperform the more naïve implementations. o0 is not shown but is super slow because it tries to perform all the work in one thread sequentially on a GPU SM which is not built to run long lists of instructions sequentially.

**Part 2 GPU**

2.2) **Ablation Study:**

A graph of different colored vertical lines

AI-generated content may be incorrect.

***Insight:*** More advanced optimizations do not always mean faster runtimes, especially in the case of cublas. On a 100-dim general matrix multiply, cublas actually performs extremely slowly compared to the more naïve optimizations. This is possibly due to it optimizing for large matrix multiplies at the cost of having a longer startup cost that cannot be offset on small calculations.

2.3 and 2.4) **Scaling Study:**

A graph of a graph

AI-generated content may be incorrect.

***Insight:***

**2.3)** O3 with the largest possible tile size 32 outperforms the other O3 variants, suggesting using the largest tile sizes is better when dealing with large matrix multiplications. Cublas is still easily the fastest on larger matrices with the naïve optimizations occupying a similar runtime space.

**2.4)** I could try using memory padding to reduce bank conflicts and loop unrolling but there is no shot I will have a faster matrix multiply than cublas.

Additional resources used:

https://chatgpt.com/share/68bcb77d-d318-8009-90f6-91fe0b35248d (Query log)

https://stackoverflow.com/questions/78739060/why-the-loop-order-influence-the-efficiency-of-code-in-gemm

https://grok.com/share/c2hhcmQtMg%3D%3D\_0977b141-1bec-4e41-8b5c-fbfe5a7753d2

CPU specs:

A screen shot of a graph

AI-generated content may be incorrect.

T4 GPU on Google Colab used.