**Computer Organization, Spring 2020**

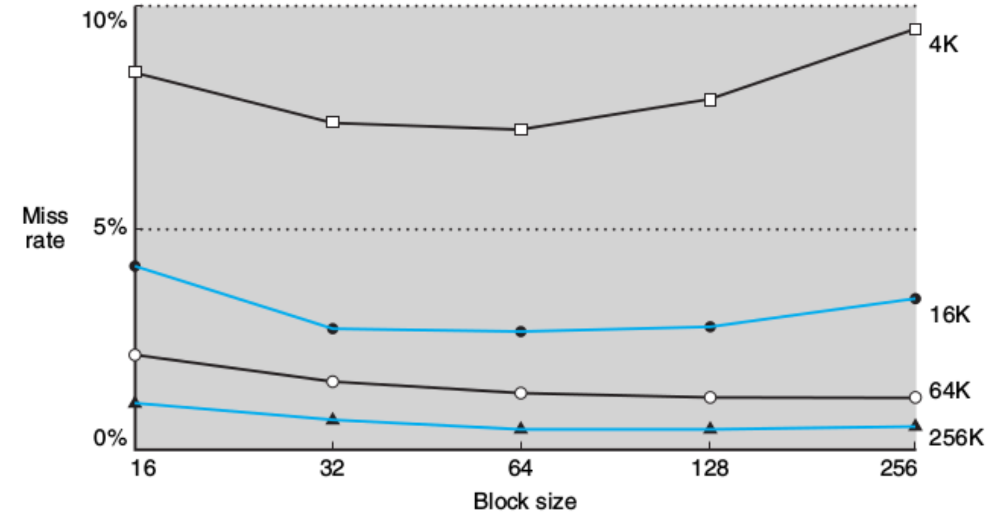
**Lab 6: Cache Simulator**

**Due: 2020/06/25**

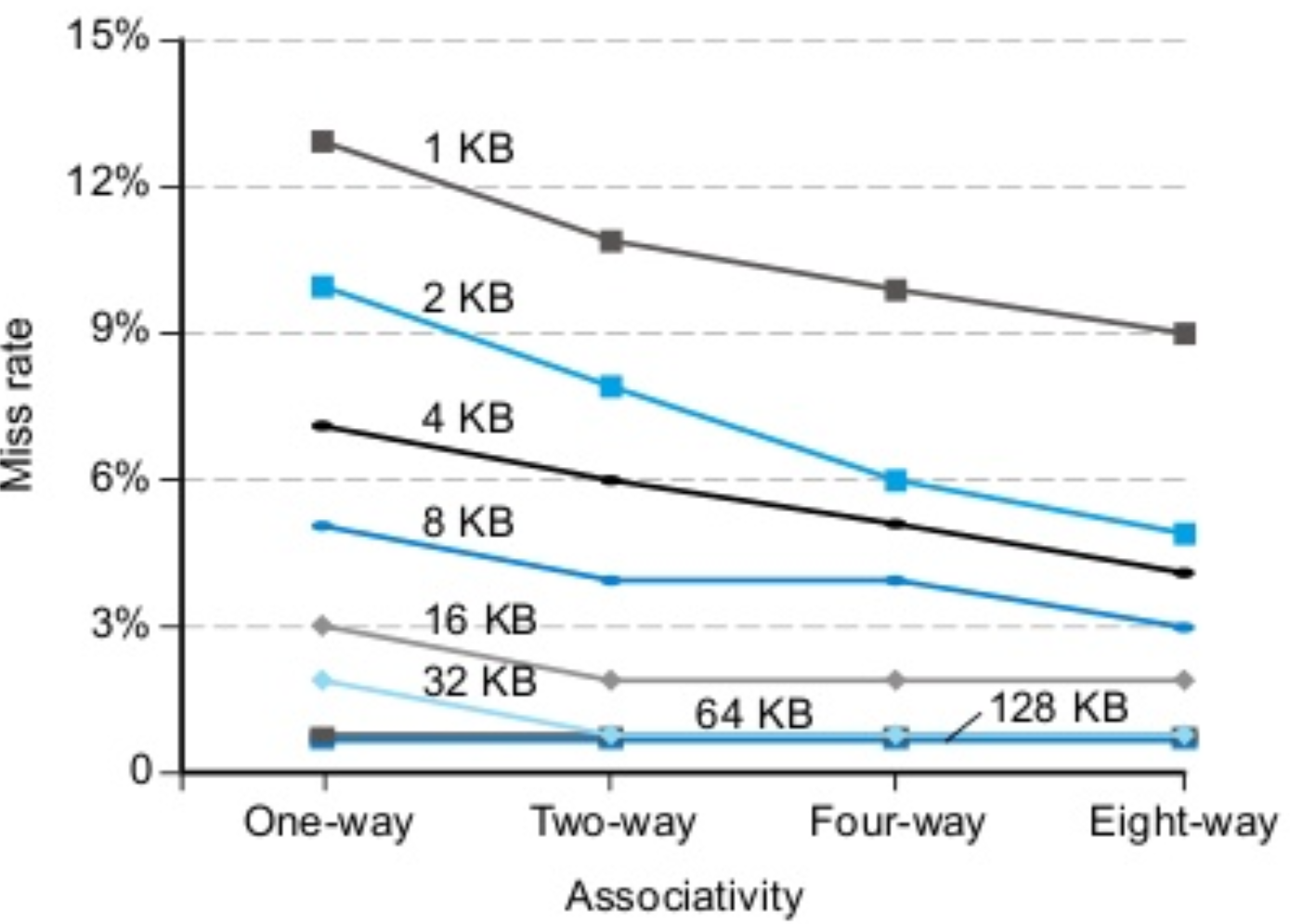
1. **Goal**  
   Cache Performance is important for system performance. In order to understand the performance difference between different cache architectures, you are asked to simulate direct mapped and n-way set associative cache behaviors by C++ style.
2. **Basic Problem (60%)**  
   **a. “ICACHE.txt” and “DCACHE.txt”:**

These 2 files are simply the memory traces of ICACHE (Instruction Cache) and DCACHE (Data Cache) respectively.

Implement a direct-mapped cache simulator and named it “direct\_mapped\_cache.cpp. Please show your output result and draw a graph as following example into your report.



1. **Advanced Problem (30%)** Implement an n-way set-associative cache simulator using LRU (Least-Recently Used) with block size = 64 bytes and named it “set\_associative\_cache.cpp”. LRU is a cache replacement policy that discards the least recently used items first. Take “LRU.txt” as inputs of the simulator and then run it. Please show your output hit-miss rate of the cache, the table below after filling it and draw your own graph as following example in your report.



Associativity

Cache Size

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1-way | 2-way | 4-way | 8-way |
| 1K |  |  |  |  |
| 2K |  |  |  |  |
| 4K |  |  |  |  |
| 8K |  |  |  |  |
| 16K |  |  |  |  |
| 32K |  |  |  |  |
| 64K |  |  |  |  |

1. **Grade**  
   (1) Basic score: 60 points,   
   (2) Advanced score: 30 points,  
   (3) Report: 10 points(use CO\_Report.docx),  
   (4) Late submission: 10 percent penalty per day,  
   (5) No plagiarism, or you will get 0 point.
2. **Hand in (will have penalty otherwise)**  
   (1) Zip your folder and name it as “GID\_ID1\_ID2.zip” (e.g. G1\_0816001\_0816002.zip) before uploading to newe3. Other filenames and formats such as \*.rar and \*.7z are NOT accepted! Multiple submissions are accepted, and the version with the latest time stamp will be graded.  
   (2) Please ONLY include C++ source codes (\*.cpp) and your report (\*.docx or \*.pdf) in the zipped folder. (Don’t need to hand in testbench.v)
3. **Q&A**  
   For any questions regarding Lab 6, please contact  
   張祐銘 yumingchang.cs03@g2.nctu.edu.tw

賴柏宏 bhbruce.cs07g@nctu.edu.tw

鄭俊賢 petertay1996.cs08g@nctu.edu.tw