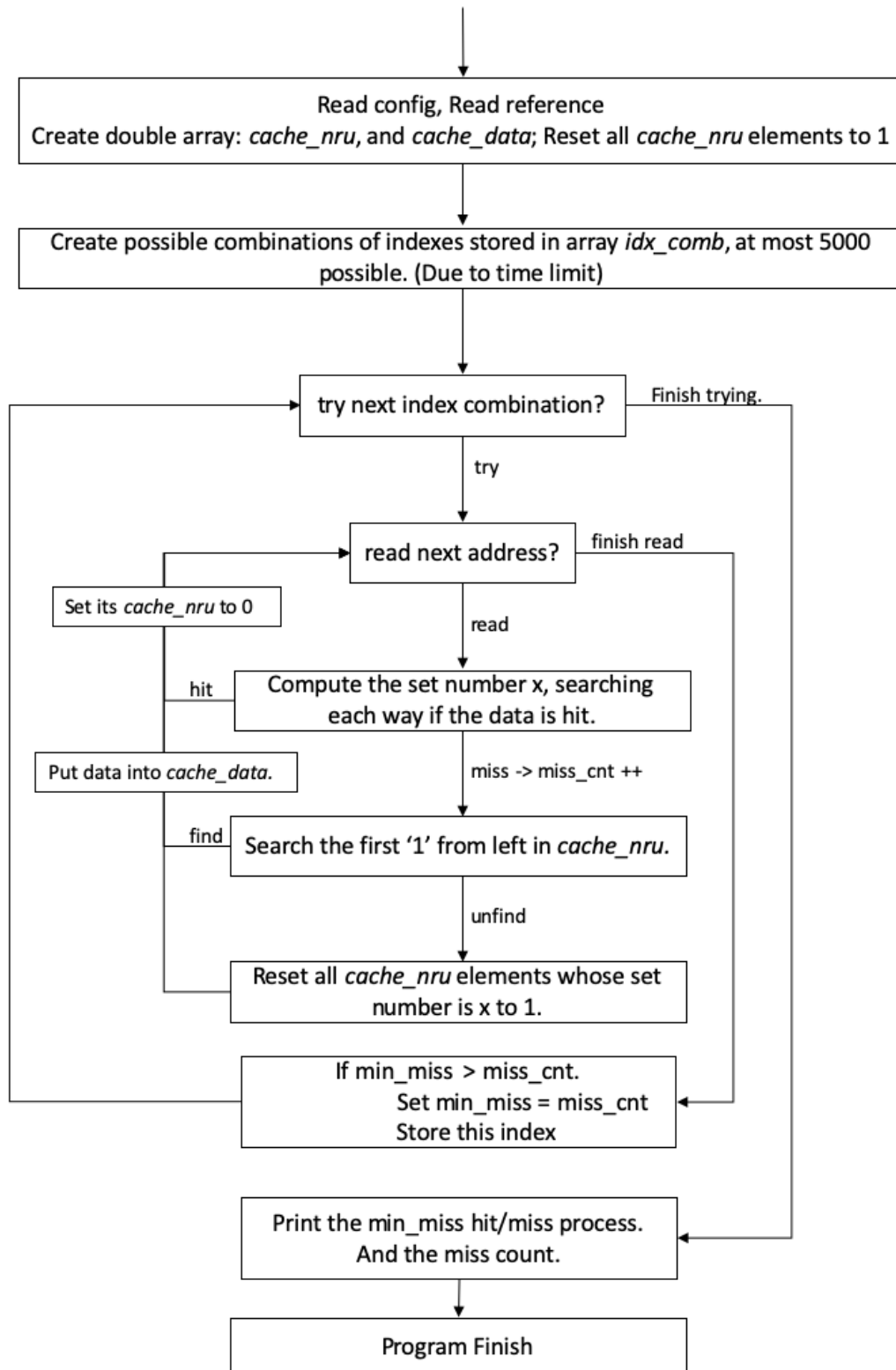


CS 410000 Computer Architecture

Final Project - NRU Replacement Policy

104021215 熊磊



這次的 Project 要實作 NRU Replacement Policy.

- 首先是把 config 和 reference 的資料讀出來。接著把計算需要的 Offset 和 Index Bits。

```
154 // Decide memory config:
155 //     M-bit address bits
156 //     B-byte block size
157 //     E cache sets
158 //     A-way set associativity.
159 //     Offset
160 //     Index Bits
161 vector<int> cache = readConfig(argv[1]);
162 vector<string> reference = readRef(argv[2]);
```

- 把所有可能的 index 組合存到 idx_comb 的 array 裡面，並用迴圈測試所有組合產生的 miss count.

```
184 idx_combination(cache[0] - cache[4], cache[5], idx_comb);
```

- 每次測試，讀下一筆 memory 的時候，用他的 index 組合去計算他的 set number。然後確認是 miss 還是 hit。如果是 hit 就把 nru_bit 設成 0。如果是 miss，則把所有 way 中 set number 一樣的 nru_bit 都設為 1，然後從第一個開始填到 cache_data 裡面，並把這個空間的 bru_bit 設成 0。Miss 時要記錄 miss count+1。
- 做完一種 index 組合的 miss count 之後，跟之前最小的 min_miss_count 做比對，如果比他還小，則把這個設為最小 miss count 得 index 組合。如果比他還大，就直接去試下一種 index 組合。

```
230 if(miss_cnt < min_miss)
231 {
232     min_miss = min(min_miss, miss_cnt);
233     min_miss_idx = z;
234 }
235 miss_cnt = 0;
```

- 做完所有 index 組合之後，把 miss count 最小的 miss/hit 過程輸出。