

CS410000 Computer Architecture
Final project: The cache behavior simulation
Name: Yun-Fan Lu Student ID: 107065507

1. Goal

Implement a cache system to simulate the behavior of cache and replacement policy.

2. Design Concept

Data Structure:

```
struct Data
{
    string reference;
    string tag;
    string str_index;
    int index;
    string offset;
    bool isMiss;
};

struct block
{
    int NRUbit = 1;
    string tag;
    string data;
};
```

Data 為 input 的 memory address，包含 tag, index, offset 等資訊。

Block 為 cache 的 block，包含 NRUbit, tag, data 等資訊。

```
vector<Data> dataList;
vector<vector<block>> cache;
```

並且使用 vector 容器。

一開始 input 時就會做 data parsing 的部分，把 offset, index, tag 等切好。丟到 dataList 中。並且先做 LSB 編碼。

之後根據 cache set 和 associativity 建立 cache system。

最後就可以開始處理資料了。

```

void processing()
{
    for(int i=0; i<dataList.size(); i++){
        int index = dataList[i].index;

        bool isFound = false;
        for(int j=0; j<cache[index].size(); j++){
            if(cache[index][j].tag == dataList[i].tag){
                cache[index][j].NRUbit = 0;
                dataList[i].isMiss = false;
                isFound = true;
                break;
            }
        }

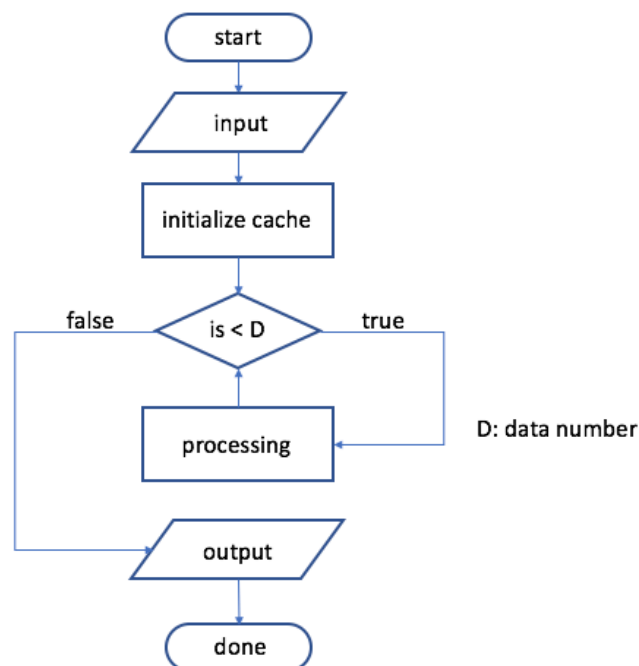
        if(!isFound){
            for(int j=0; j<cache[index].size(); j++){
                if(cache[index][j].NRUbit == 1){
                    dataList[i].isMiss = true;
                    cache[index][j].NRUbit = 0;
                    cache[index][j].tag = dataList[i].tag;
                    break;
                }
                else if(cache[index][j].NRUbit == 0 && j == cache[index].size()-1){
                    dataList[i].isMiss = true;
                    cache[index][0].NRUbit = 0;
                    cache[index][0].tag = dataList[i].tag;

                    for(int k=1; k<cache[index].size(); k++){
                        cache[index][k].NRUbit = 1;
                    }
                    break;
                }
            }
        }
    }
}

```

根據 LRU policy 實作。根據 index 欄位找資料，如果 tag 相符，紀錄 hit 且 NRUBit 設為 0。如果找不到相符的資料，找一 NRUBit 為 1 的 block 放入 data，如果所有 block 皆為 0，則將全部 block 設為 1，再重新開始。

Flow chart



3. Result

結果經 `verify` 測試皆正確。