

CSC3150 Operating System

Assignment Report #3

Name: Yang Yin

Student ID: 120090516

Date: 2022.11.9

The Chinese University of Hong Kong, Shenzhen

1. Environment

The cluster is used to solve the assignment.

Cluster Environment

The following information holds for every machine in the cluster.

Item	Configuration / Version
System Type	x86_64
Operating System	CentOS Linux release 7.5.1804
CPU	Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GHz 20 Cores, 40 Threads
Memory	100GB RAM
GPU	Nvidia Quadro RTX 4000 GPU x 1
CUDA	11.7
GCC	Red Hat 7.3.1-5
CMake	3.14.1

Fig. 1 cluster environment

2. Execution steps

- (1) Use vscode to open the project
- (2) Enter sbatch slurm.sh in terminal.
- (3) Wait for the result.

3. Design

Functions:

- (1) init_invert_page_table: Initialize invert_page_table.
- (2) vm_init: initialize the virtual memory.
- (3) vm_read: return the value of address(addr) in vm.
- (4) vm_write: change the value of address(addr) in vm to value.
- (5) vm_snapshot: output the input_size data to result with the address offset value

(6) `vm_find_phy_addr`: return the physical address with the virtual address.

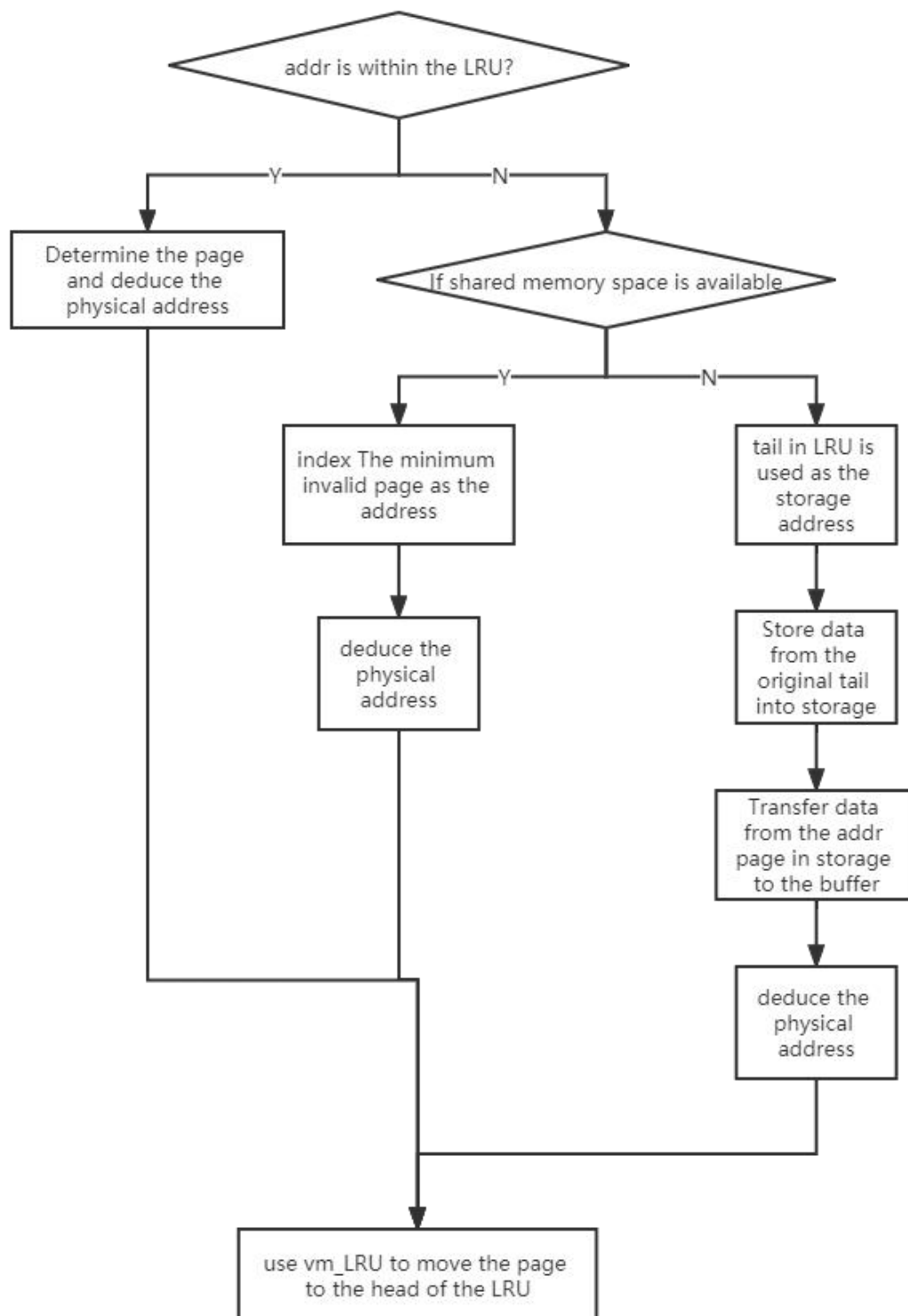


Fig. 2 flowchart of `vm_find_phy_addr`

(7) vm_LRU: Move the page to the head position in the LRU

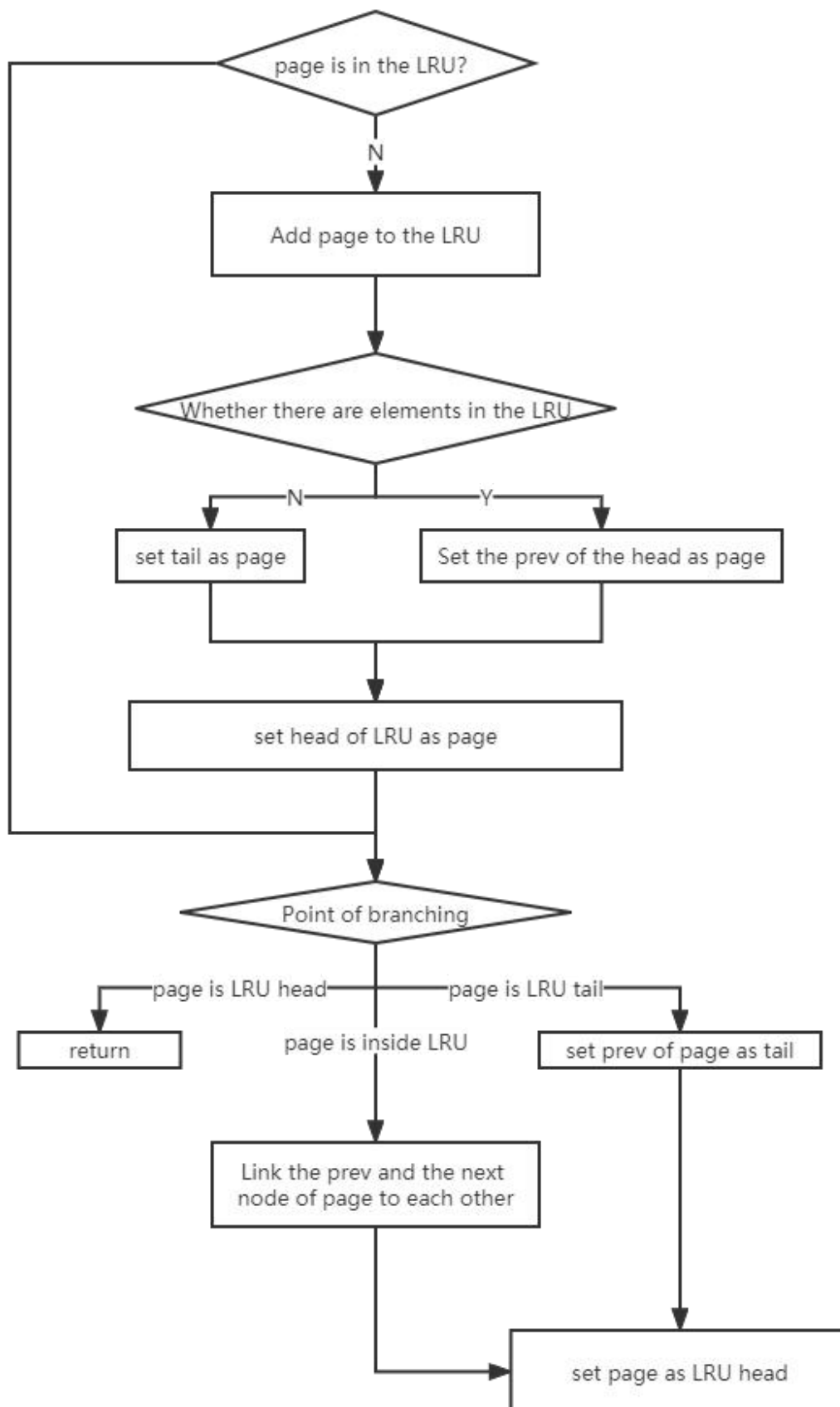


Fig. 3 flowchart of vm_LRU

4. Screenshot

```
input size: 131072
pagefault number is 8193
```

Fig. 4 out put of case 1

```
__device__ void user_program(VirtualMemory *vm, uchar *input, uchar *results,
                             int input_size) {
    for (int i = 0; i < input_size; i++) //input_size 128KB 4*(32*1024)
        vm_write(vm, i, input[i]); //4096 pages, all new to LRU, page_fault+=4096

    for (int i = input_size - 1; i >= input_size - 32769; i--) //input_size-1 to input_size - 32768 is in LRU
        int value = vm_read(vm, i); //page_fault+=1 (input_size-32769)

    vm_snapshot(vm, results, 0, input_size); //input_size 128KB 4*(32*1024)
    //4096 pages, as LRU pages is update, all new
}
```

Fig. 5 code of case 1

```
input size: 131072
pagefault number is 9215
```

Fig. 6 output of case 2

```
__device__ void user_program(VirtualMemory *vm, uchar *input, uchar *results,
                             int input_size) {
    // write the data.bin to the VM starting from address 32*1024
    for (int i = 0; i < input_size; i++) //4096 pages
        vm_write(vm, 32*1024+i, input[i]);
    // write (32KB-32B) data to the VM starting from 0
    for (int i = 0; i < 32*1023; i++) //1023 pages
        vm_write(vm, i, input[i+32*1024]);
    // readout VM[32K, 160K] and output to snapshot.bin, which should be the same with data.bin
    vm_snapshot(vm, results, 32*1024, input_size); //4096 pages (LRU is 0-1022, 5120)
}
```

Fig. 7 code of case 2