# Project 5 Virtual Memory Manager

# The Implementation of LRU in TLB

The function prototype of the LRU replacement:

Old Version:

int TLB\_replacement\_LRU(page\_t new\_p\_num, frame\_t new\_f\_num, tlb\_t tlb);

New Version:

# Put It All Together: Address Translation

The functions to be integrated include:

/\* see function 3 in Lec14c-Project 5 Data Structures Exercise Handout \*/

int search\_TLB(page\_t p\_num, tlb\_t tlb, bool \*is\_tlb\_hit, frame\_t \*f\_num)

/\* see function 4 in Lec14c-Project 5 Data Structures Exercise Handout \*/

int search\_page\_table(page\_t p\_num, page\_table\_t p\_table, bool \*is\_page\_fault,

frame\_t \*f\_num);

/\* see function 5 in Lec14c-Project 5 Data Structures Exercise Handout \*/

int page\_fault\_handler(page\_t p\_num, physical\_memory\_t \*physical\_mem,

page\_table\_t \*p\_table, tlb\_t \*tlb);

/\*

\* Get a page number from a logical address.

\* Input: a logical address.

\* Output: a page number

\*/

int get\_page\_num(logic\_address\_t l\_addr, page\_t \*p\_num);

page\_t page\_num;

logic\_address\_t logic\_address;

logic\_address\_loader(logic\_address\_file\_name, logic\_address\_list);

get\_a\_logical\_address(logic\_address\_list, logical\_address);

get\_page\_number(

search\_TLB(page