Faculty of Computer Science & Engineering

Operating Systems

Contact:
Trần Ngọc Anh Tú
51304672@hcmut.edu.vn

Ho Chi Minh City
University of Technology

Lab 8 - Memory Management

Objective

- Understand fragmentation and know how to handle it.
- Know Paging techniques.

Fragmentation

- Exercise: Compile and execute the following program (remember to set ulimit –v 262144)

```
#include <stdio.h>
#include <stdlib.h>
int * a [100000]
int main() {
    int i;
    for (i = 0; i < 100000; i++) {
        if ((a[i] = (int*)malloc((1 << 10) * sizeof(int))) == NULL) {
            break;
        }
    }
    printf("Created %d 4KB memory region(s)\n", i);
    free(a[0]), free(a[2]);
    int * b = (int*)malloc((1 << 11) * sizeof(int));
    if (b == NULL) {
        printf("Cannot allocate more memory\n");
    }else{
        printf("Allocated a new 8KB memory region\n");
    }
}
```

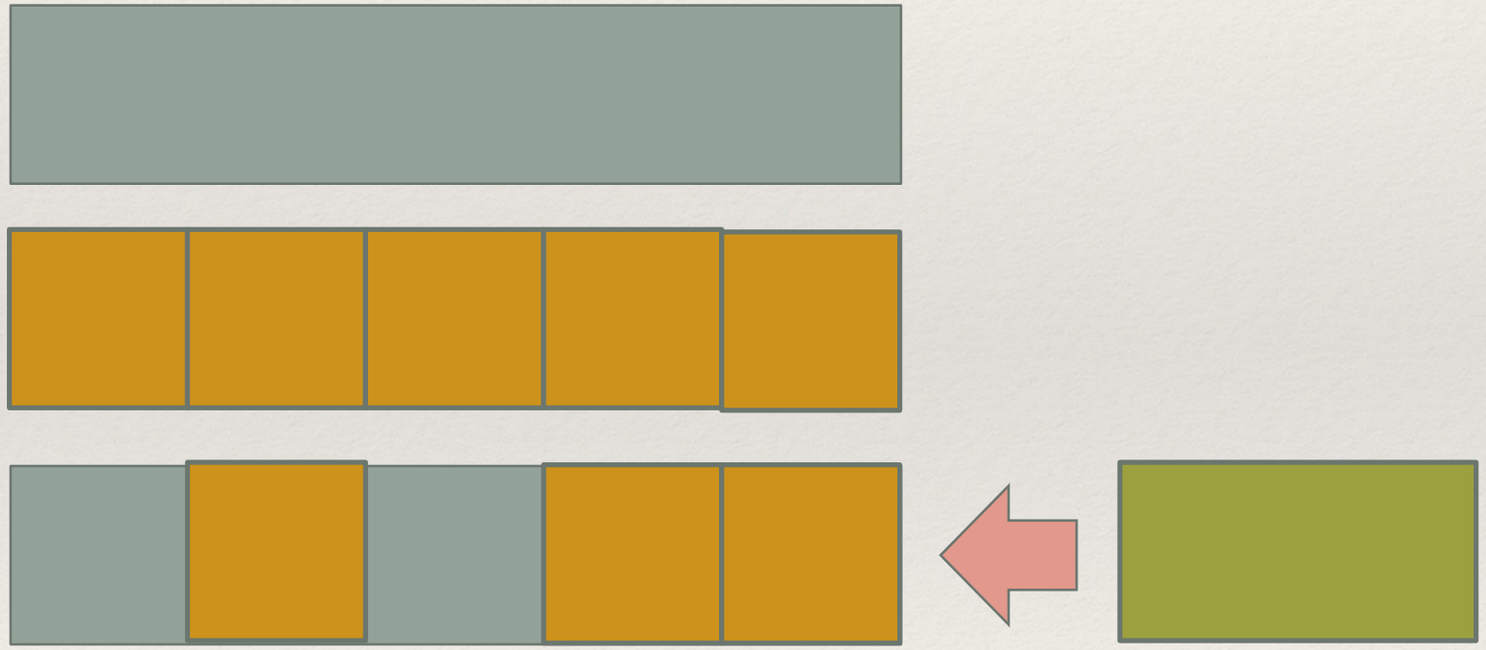
Fragmentation

- Exercise: Compile and execute the following program (remember to set ulimit –v 262144)

```
#include <stdio.h>
#include <stdlib.h>
int * a [100000]
int main() {
    int i;
    for (i = 0; i < 100000; i++) {
        if ((a[i] = (int*)malloc((1 << 10) * sizeof(int))) == NULL) {
            break;
        }
    }
    printf("Created %d 4KB memory region(s)\n", i);
    free(a[0]), free(a[1]);
    int * b = (int*)malloc((1 << 11) * sizeof(int));
    if (b == NULL) {
        printf("Cannot allocate more memory\n");
    }else{
        printf("Allocated a new 8KB memory region\n");
    }
}
```

Fragmentation

- Fragmentation is a phenomenon in which the memory space is used inefficiently.



Fragmentation

- There is two common types of fragmentation: internal fragmentation and external fragmentation.
- Both OS and application are responsible to avoid fragmentation.
  - Slab allocator
  - Standard memory allocator functions

Memory placement

- Fixed Partitioning
- Dynamic Partitioning
  - Placement policies
    - First fit
    - Best fit
    - Next fit
    - Worst fit

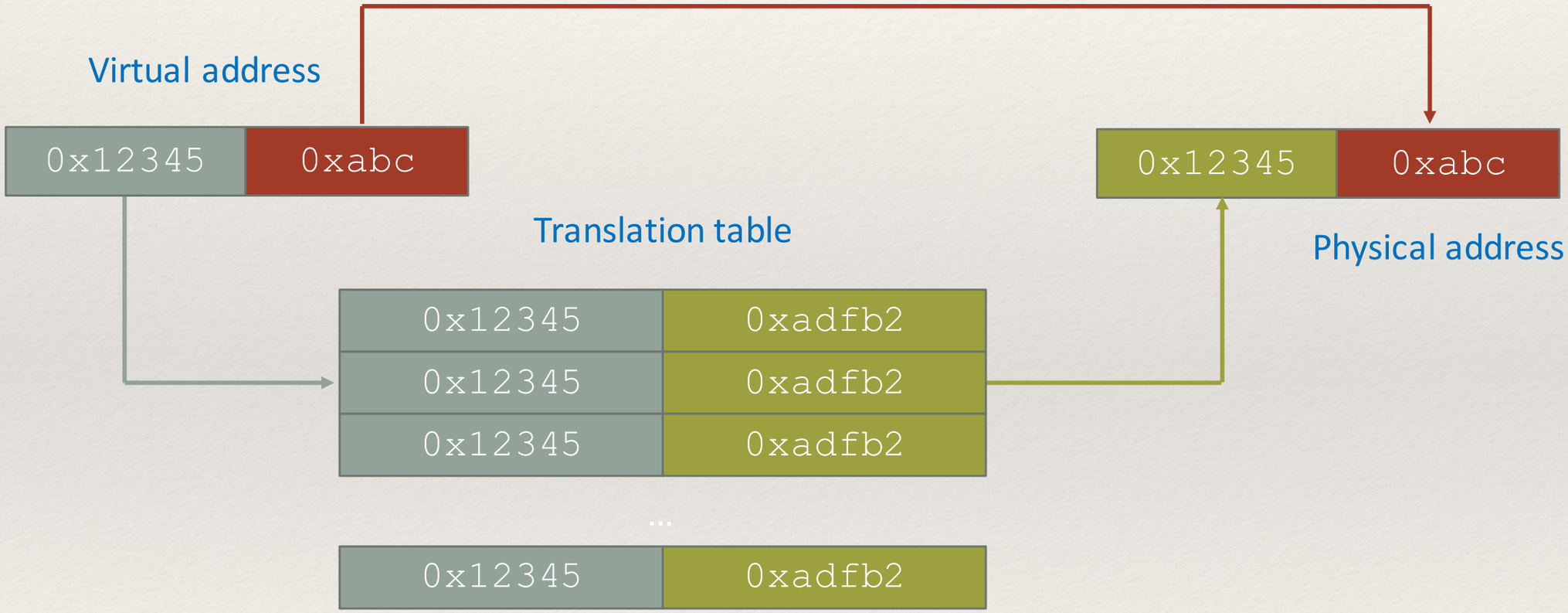
Virtual memory

- Do you remember those?
  - Page
  - Page table
  - Memory Management Unit (MMU)
  - Swapping
  - Thrashing

Virtual memory

- The memory address space is divided into multiple blocks of contiguous virtual memory addresses called pages.
  - In Linux, the default page size is 4KB.
- Page tables are used to translate the virtual addresses seen by processes to physical addresses used by the hardware to process instructions.
- Translation process is often handled by Memory management unit (MMU).
- If the amount of memory processes require is greater than that of main memory, some pages are swapped out to secondary storage to make rooms for the new pages. Those pages are then swapped in when processes need them again.
- Thrashing occurs when pages are continuously swap in and out memory.

Virtual memory



Virtual memory

- Exercise: Compile and run the following program. Remember to execute ulimit –v unlimited first and use free to check the amount of free memory.

```
#include <stdio.h>
#include <stdlib.h>
#define SIZE (1 << 28)
int main() {
    while (1) {
        malloc(SIZE);
    }
}
```

```
gcc <source_code> -o swap
./swap &
free
```

Virtual memory

- Exercise: Do address translation exercise.

End

Thanks!