Operating Systems
Dr. Youssef Iraqi
Assignment 2: Memory Management

Introduction

The main goal is to give the participating students an empirical introduction to aspects of memory management. This will serve as a complement to the theoretical baseline of the lectures.

Problem Statement

You are requested to write a Java (or a language of your choice) program of a simple Memory Management Unit.

The program should allow the following:

- The user can create a process asking for memory. The program will return a process ID if the requested memory can be allocated. It will also print the allocated Base and Limit.
- The user can delete a process by specifying a process ID. The program should do that and free any allocated memory for the deleted process.
- The user can, using a process ID, ask to convert between virtual addresses and physical addresses.
- The user can ask the program to print the memory map showing what memory is allocated and to which process.

Your program should be named `mmu` and should accept two arguments specifying the amount of memory to be managed in KB and the memory allocation strategy (1: First Fit, 2: Next Fit, 3: Best Fit, and 4: Worst Fit).
e.g.

```
java mmu 50000 3
```

Your program should display a prompt for the user and should accept the following **4** commands:

1. First

```
cr AMOUNT_OF_REQUESTED_MEMORY
```

Description:
Create a process and allocate the requested amount of memory (in KB) to it.
The command should return the process ID and the Base and Limit of the allocated memory. The command may return an error message if there is not enough memory.
e.g.

```
> cr 1500
```

2. Second

```
> dl PROCESS_ID
```

Description:
Delete the specified process and free the allocated memory.
The command should return an error message if there is no process with the specified ID.
e.g.

```
> dl 6
```

3. Third

```
> cv PROCESS_ID VIRTUAL_ADDRESS
```

Description:
Make a conversion for the specified process ID from the specified Virtual Address to the Physical Address.
The command should return an error message if the process tries to access an address outside its address space.
e.g.

```
> cv 6 200
```

4. Fourth

```
> pm
```

Description:
Print the memory map.
The command should print which memory locations are assigned and to which processes, and which memory locations are free.

Deliverables:

- You should design your program in a modular way. Especially you should have a module that decides where to allocate memory for a new process.
- Use all what you have learned in software engineering.
- You should devise a sequence of tests that show that your program works correctly. Construct your test cases so they are relatively compact but test your program reasonably well. Your test cases should be included in your report.
- You should deliver a softcopy of both the source code and the report.

Note: You can make reasonable assumptions and explain them in your report.

Rules, Regulations, and Grading Policy:

The assignment is to be done in groups of two students (except for one group as it was the case for Assignment 1). 70% of the grade is for the correctness and understanding of the solution. The other 30% are for documentation and programming efficiency. Present your work using a demonstration in class and submit soft copies of the codes and sample outputs. This assignment should be submitted by the deadline indicated on Canvas. 10% will be deducted for any one-day (full or partial) delay. You will receive extra marks for relevant extra work.