SFWRENG 3SH3 Operating Systems

by Dr. Anwar M. Mirza
Lab Demo: During Lab time of week starting from March 14th, 2022
Lab Report Due: March 20th, 2022, 11:59PM

Lab Assignment 03

Date: March 8th, 2022

Main Memory Management

1. Background

In this assignment you will explore and test various contiguous memory allocation schemes. The lab assignment is based on the material covered in Section 9.2 and the programming project P-49 (Chapter 9) of the textbook.

2. Assignment

Q.No. 1. Write a C/C++ program to manage a contiguous region of memory of size `MAX`, where the addresses range from `0` to `MAX - 1`. Write appropriate functions to cater for the:

1) Request for a contiguous block of memory – `function allocate(…){ … }`
2) Release of a contiguous block of memory – `function release(…){ … }`
3) Compaction of unused holes of memory into one single block – `function compact(…){ … }`
4) Report the regions of free and allocated memory – `function status(…){ … }`

The program should be able to allocate memory using all of the following three approaches:

I. First fit
II. Best fit
III. Worst fit

and do the following:

a) Initialization with 1MB (1,048,576 bytes) of free memory at the start.
b) Fill up the 1MB memory using the allocation requests, each being an integer multiple of 4KB (4096 bytes) taken from the range 4KB to 100KB. Maintain a list of allocations (processes) currently in memory.
c) Remove 10% of the processes randomly from the memory (the number of processes to be released, is the rounded integer closest to 10% of processes currently in the memory). For example, if currently there are 64 processes in the memory, then remove `round(64/10) = 6` processes from random locations (from the list of allocations) in memory. This should create free spaces (holes) in the memory.
d) Display the status of the memory (regions of free spaces (holes) and filled spaces (processes) in memory).
e) Do this part without compaction. Using the three allocation appeaoches, fill in the free spaces (holes) using new allocation requests generated in the same way as in part (b), and display the status of the memory for each one of the three allocation approaches.
f) Now repeat part (e) using compaction.
g) Comment on which of the three approaches (first fit, best fit, worst fit) is better in memory utilization, with and without compaction.

Dr. Anwar M. Mirza, Department of Computing and Software
Faculty of Engineering, McMaster University
Hamilton, Ontario, Canada

3. Guideline

- Work in your already allocated teams.
- You are recommended to use the Lab hours to carry out this work. TAs will be available during their respective Lab sessions for your in-person / online help.
- For this lab, you need to
    a. Give a lab demo during the time slot allocated to your Lab group by the TA, during the week starting from March 14th, 2022.
    b. Submit / upload a group Lab report in MS Word or PDF by March 20th, 2022, 11:59PM.
- Copying/reproduction of (all or parts of) reports from other groups will lead to zero marks for the whole team.

Please use MS Teams Lab Q&A and Lab Section channels to coordinate your work with your team members and with the TAs.