

Laporan Praktikum Deadlock

Praktikum Sistem Operasi PTI Kelas C

Nama Mahasiswa:

Ahmad Nauval Syahputra 215150601111021

Asisten Praktikum:

Iqbal Biondy 205150601111009

Dosen:

Faizatul Amalia, S.Pd.,M.Pd



Program Studi Pendidikan Teknologi Informasi Jurusan Sistem Informasi Universitas Brawijaya

2022

Ahmad Nauval Syahputra 215150601111021

1. Langkah Praktikum

- a. Isikan Langkah-langkah dan screenshoot sesuai dengan modul praktikum
- b. Setiap screenshoot wajib disertai nama dan nim
- 2. Latihan Praktikum
 - a. Isikan Latihan Praktikum dan screenshoot sesuai dengan modul praktikum
 - b. Setiap screenshoot wajib disertai nama dan nim
- 3. Kesimpulan

Buatlah dalam bentuk paragraph dalam menjawab pertanyaan ini

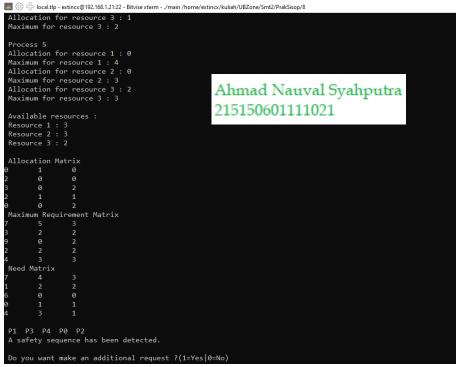
- a. Jelaskan hubungan antara proses dan thread
- b. Jelaskan perbedaan dari singlethread dan multithread
- c. Sebutkan dan jelaskan macam-macam multithreading model
- 4. Referensi

Buatlah referensi dengan format sesuai dengan referensi pada penulisan skripsi http://file-

filkom.ub.ac.id/fileupload/assets/upload/filemanager/Skripsi/PanduanSkripsiFilkom-v3.0.pdf>

- 1. Langkah Praktikum
 - a. Login ke system Linux/GNU kemudian buka terminal
 - b. Kompilasi program dengan frontend clang

- c. Jalankan program dengan command ./program
- d. Input Data pada program sesuai ketentuan



e. Amati hasilnya.

Input proses yang dimasukkan pada matriks program kemudian program mengurutkan queue dengan safety sequence sehingga

urutan prosesnya adalah P1-P3-P4-P0-P2 agar tidak terjadi deadlock pada runtime prosesnya.

2. Latihan Praktikum

A. Jalankan ulang dengan tambahan P5 menambah R1=3, R2=3 , R3=0.

Dengan menambah request tambahan pada Proses 5 dengan alokasi request ressource 1 sejumlah 3, ressource 2 sejumlah 3, dan ressource 3 sejumlah 0. Deadlock terjadi karena ressource yang diminta (tambahan requestnya) tidak terpenuhi saat proses berjalan dan algoritma banker tidak mampu mengatasinya.

Sistem Operasi : Deadlock

B. Jalankan ulang dengan tambahan P5 menambah R1=0, R2=2, R3=0.

Setelah menambah request pada Proccess 5 dengan request ressource 1 sejumlah 0, ressource 2 sejumlah 2, dan ressource 3 sejumlah 0. Sehingga memicu safety sequence sehingga tidak terjadi deadlock. Dengan ururtan program P3-P4-P1-P2-P0 maka kondisi sistem tetap pada safety state.

3. Kesimpulan

Diperlukannya algoritma dalam mengatasi adanya deadlock, tentunya algoritma pemrograman selalu berkembang sampai sekarang. Algoritma banker merupakan algoritma umum yang di implementasikan pada sistem komputer, dengan mengatasi request proses dan mengurutkannya kedalam sequence queue yang benar agar tidak terjadi deadlock, namun apabila ressource yang diminta terlalu berlebihan dalam kasus tersebut masih mungkin terjadi deadlock.