

Data de Entrega:

O que deve ser entregue:

1. Deve ser enviado por email um arquivo zipado contendo:
 - a. os códigos fontes dos programas implementados e a bateria de testes utilizada durante a execução do programa.
 - b. O programa deve ser apresentado para o monitor no dia da entrega

Trabalho Pratico 6 – Algoritmos Safety, Avoid e Detection

Neste laboratório você deve implementar 3 algoritmos relacionados com o gerenciamento de deadlock. Os algoritmos que devem ser implementados devem ser; Safety (verificar se o sistema está em estado seguro); Avoid (no caso de uma solicitação de recurso por um processo) e Detection (verificar se sistema entrou em deadlock). Para implementar esses algoritmos, você deve receber como entrada um conjunto de processos, uma quantidade de tipos de recursos, as necessidades de recursos dos processos e um estado inicial de solicitação de recursos pelos processos:

1. (3,0 pontos) implemente o algoritmo *Safety* para verificar se existe uma sequência de execuções dos processos segura. Utilize como dados para a execução de seu algoritmo o seguinte exemplo. Crie outros exemplos para validar seu algoritmo.

	Alocado no momento t0			Máximo requerido pelo processo		
	A	B	C	A	B	C
P0	3	1	0	7	5	7
P1	7	0	1	7	1	1
P2	0	2	2	10	4	3

2. (3,0 pontos) Dada uma nova solicitação de recurso para um processo, implemente o algoritmo *Avoid* para verificar se é possível permitir esta solicitação deste recurso, evitando um deadlock. Caso seja detectada a possibilidade do sistema entrar em deadlock, então o processo deverá ser bloqueado.
3. (2,5 pontos) Implemente o algoritmo *Detection* para detectar a existência de deadlock em um conjunto de processos em execução. O algoritmo deve imprimir o numero do processo em deadlock. A partir do exemplo de dados apresentado acima crie um conjunto de dados que gere um deadlock.