

1.

a. Enuncie os principais objectivos de um Operativo.

Executar programas do utilizador e tornar mais fácil a resolução de problemas.

Tornar fácil o uso da máquina.

Utilizar o hardware do computador duma forma eficiente

b. Diga o que entende por:

i. Processo.

Um processo é basicamente um programa em execução, sendo este constituído pelo código executável e respectivos dados, pilha de execução, contador de programa, valor do apontador da pilha, valores dos registadores de hardware, além do conjunto de outras informações necessárias à execução do programa.

ii. Tabela de processos.

contém informação sobre o estado dos processos. Cada processo tem um identificador pid, um *proprietário*, uma *prioridade*, memória atribuída, ficheiros abertos, o estado, etc... ps comando do UNIX que fornece informação sobre os processos em curso.

iii. Multi-programação.

Divide-se a memória em diversas partes com um Job alocado em cada uma delas, enquanto um Job espera a conclusão da sua operação I/O, um outro Job pode estar a utilizar o processador, a ideia é manter na memória simultaneamente uma quantidade de jobs suficiente para ocupar o processador a 100% e aproveitar ao máximo.

2.

a. Quais os objectivos da Comunicação entre Processos.

Garantir a integridade e a confiabilidade na execução de aplicações concorrentes

b. Complete o pseudo-código apresentado, escrevendo as instruções correctas (na folha de resposta) relacionando-as com as etiquetas [i), i i), i i i)] das caixas em branco conforme apresentado na figura abaixo.

```
#define CHAIRS 5
```

```
typedef int semaphore;
```

```
Semaphore customers = 0;
```

```
Semaphore barbers = 0;
```

```
Semaphore mutex = 1 )
```

```
Int waiting = 0;
```

```
Void barber(void){
```

```
    While(TRUE){
```

```
        down(&customers);
```

```
        down( &mutex)
```

```
        waiting = waiting -1;
```

```
        up(&barbers);
```

```
        up(&mutex);
```

```
        cut_hair();
```

```
    }
```

```
}
```

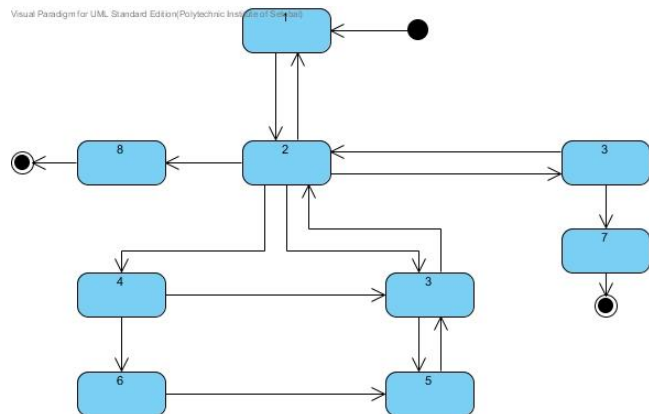
```

Void customer(void){
    Down(&mutex);
    If(waiting < CHAIRS){
        Waiting = waiting +1;
        up(&costumers);
        up(&mutex);
        down(&barbers);
        get_haircut();
    }else{
        up(&mutex);
    }
}

```

3. Analisando o diagrama de estados dum processo, conforme apresentado na figura abaixo, explique em que consiste cada um dos estados apresentados (8 estados no total).

- 1-Execução em modo utilizador
- 2-Execução em modo nuclear (Kernel)
- 3-Pronto para executar
- 4-Bloqueado
- 5-Pronto, em memória secundária
- 6-Bloqueado, em memória secundária
- 7-Criado
- 8-Extinção



- 1-Execução em modo utilizador
- 2-Execução em modo nuclear (Kernel)
- 3-Pronto para executar
- 4-Bloqueado
- 5-Pronto, em memória secundária
- 6-Bloqueado, em memória secundária
- 7-Criado
- 8-Extinção

4. Justifica as seguintes afirmações:
- a. Os segmentos têm de ser blocos de memória de tamanho variável;
 - b. Através de paginação é possível obter um espaço de endereçamento (virtual) maior do que o espaço de endereçamento que a memória RAM instalada no computador permite.