Questionário de Sistemas Operacionais

Gustavo Soares – 3°B

1. Quais as principais partes de um processo?

Em geral, processos de sistemas computacionais são formados pelos seguintes recursos:

* Uma imagem do código de máquina executável associado com um programa.
* Memória, que inclui o código executável, dados específicos do processo (dados da entra e saída), uma chamada de pilha (para manter a pilha de [sub-rotinas](https://pt.wikipedia.org/wiki/Sub-rotina_aninhada) ativas e/ou outros eventos) e um heap (alocação dinâmica de memória) para manter a computação intermediária gerada durante o tempo de execução.
* Descritores de sistema operacional que são alocados aos processo, como os descritores de arquivo do (segundo a terminologia UNIX) ou [handles](https://pt.wikipedia.org/wiki/Handle_(inform%C3%A1tica)) (usado no [Windows](https://pt.wikipedia.org/wiki/Microsoft_Windows)) e os códigos e sinks.
* Atributos de segurança, tais como conjunto de permissões para o usuário que criou o processo, são as operações permitidas.
* Contexto de estado do processador, tal como o conteúdo dos registradores, memória física de endereçamento. O estado é normalmente guardado nos registradores quando o processo está em execução, e na memória principal caso contrário.

1. O que é um ambiente monothread?

Em um ambiente Monothread, uma aplicação é composta por códigos que são executados sequencialmente, de acordo com repetições, desvios, funções e procedimentos, onde a utilização de concorrência é efetuada através da implementação de processos independentes e subprocessos.

1. Caracterize um ambiente multithread.

Em um ambiente Multithread, o conceito de thread é amplamente explorado e aplicado, onde técnicas permitem que as mesmas sejam executadas de maneira simultânea. Através da utilização deste recurso, os softwares podem efetuar diversas tarefas de maneira paralela, possibilitando assim que as threads exerçam suas funções de maneira independente umas das outras.

1. Qual a vantagem no uso de threads?

Suas vantagens são o aumento de desempenho com a exploração do paralelismo real, melhor capacidade de resposta - a criação de uma thread é mais rápida do que a criação de um novo processo - e economia de recursos, já que os recursos alocados para o processo podem ser compartilhados entre as threads.