F6.Processamento: Paralelismos

Entrega Sem prazo F
Limite de tempo Nenhum

Pontos 1 Perguntas 12

Tentativas permitidas Sem limite

Fazer o teste novamente

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MANTIDO	Tentativa 6	Menos de 1 minuto	1 de 1
MAIS RECENTE	Tentativa 6	Menos de 1 minuto	1 de 1
	Tentativa 5	Menos de 1 minuto	0,92 de 1
	Tentativa 4	Menos de 1 minuto	0,92 de 1
	Tentativa 3	Menos de 1 minuto	0,75 de 1
	Tentativa 2	Menos de 1 minuto	0,58 de 1
	Tentativa 1	4 minutos	0,42 de 1

(!) As respostas corretas estão ocultas.

Pontuação desta tentativa: 1 de 1

Enviado 30 mai em 20:46

Esta tentativa levou Menos de 1 minuto.

Pergunta 1

0,08 / 0,08 pts

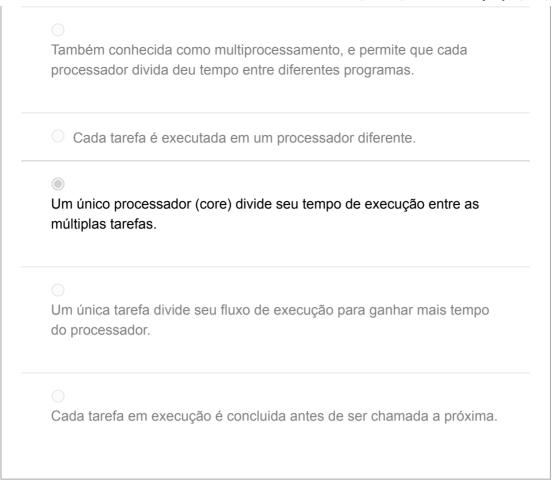
{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/TnOu-IIBmtc)

e responda} O que entendemos como Multitasking (Multitarefa)?



Pergunta 2

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/TnOu-IIBmtc)

e responda} O que ocorre em um sistema operacional Multitask rodando em um processador com vários núcleos?



As tarefas são divididas entre os diferentes núcleos. Cada núcleo realiza Multitasking para executar as tarefas a ele atribuídas.

Cada tarefa em execução é concluída antes de ser chamada a próxima.

Há uma fila de tarefas para cada processador.

Cada tarefa é executada em um processador diferente.

Uma mesma tarefa é executada por diferentes núcleos.

Uma tarefa só pode ser completada se outra for completamente concluída por outro núcleo.

Pergunta 3

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} O que entendemos por Preenptive Multitasking (Multitarefa Preemptivo)?

O sistema operacional concede um tempo de execução a uma tarefa e esta processa até concluir seu processamento, evitando ser interrompida.

O sistema operacional previamente seleciona qual a tarefa com maior prioridade.

O sistema operacional concede um tempo de execução para uma tarefa e retira arbitrariamente.

Significa que a tarefa possui alta prioridade e, com o passar do tempo, o sistema começa a reduzir sua prioridade, permitindo que tarefas menores tenham maior prioridade.

É uma fila de tarefas em espera para execução.

Pergunta 4

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} O que entendemos por Multithreading?

O sistema operacional previamente seleciona qual a tarefa com maior prioridade.

É um conceito semelhante a Multitasking, porém a mesma tarefa se subdivide em múltiplas linhas de execução.

Significa que a tarefa possui alta prioridade e, com o passar do tempo, o sistema começa a reduzir sua prioridade, permitindo que tarefas menores tenham maior prioridade.

É uma fila de tarefas em espera para execução.

O sistema operacional concede um tempo de execução a uma tarefa e esta processa até concluir seu processamento, evitando ser interrompida.

Pergunta 5

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} Qual as principais diferenças entre Multitasking e Multithreading?



Em Multithreading, as Threads compartilham o mesmo espaço de memória, conexões de rede, arquivos, etc. Em Multitasking, as tarefas não compartilham estes recursos.

Em Multithreading, as Threads não podem compartilhar o mesmo core. Em Multitasking, as Tasks podem ser executadas no mesmo core.

Em Multithreading, as Threads não compartilham o mesmo espaço de memória, conexões de rede, arquivos, etc. Em Multitasking, as tarefas compartilham estes recursos.

Não há diferenças entre Threads e Tasks

Em Multithreading, a prioridade atribuída a uma Thread precisa ser maior que a atribuída a uma Task.

Pergunta 6

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} O que dizer sobre a execução de Multithreading em um processador com um único núcleo (core)?

Significa que a Thread possui alta prioridade e, com o passar do tempo, o sistema começa a reduzir sua prioridade, permitindo que Threads menores tenham maior prioridade.

Acaba funcionando exatamente igual a Multithask

O processador divide seu tempo de execução entre as diversas Threads.

Não é possível realizar Multithreading em um processador com um único núcleo.

Como possui apenas um núcleo, o sistema operacional concede um tempo de execução a uma tarefa e esta processa até concluir seu processamento, evitando ser interrompida.

Pergunta 7

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} O que podemos afirmar sobre Multiprocessing (multiprocessamento)?

- Em Multiprocessing n\u00e3o h\u00e1 necessidade de termos Multithreading.
- Multiprocessing só é permitido em supercomputadores.
- Não existe Multiprocessing em um processador de núcleo único.
- Em Multiprocessing não há necessidade de termos Multitasking.
- Assim como em Multitasking, em Multiprocessing não é possível executar dois programas simultaneamente.

Pergunta 8

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} No instante 5:00 o vídeo inicia um teste usando um Raspberry Pi Zero. Por que razão a CPU nunca superou os 100% ao executar os três programas de teste?

O Raspberry Pi Zero não possui Multithreading, garantindo que a CPU nunca alcance seu limite de execução.

Como o Raspberry Pi é um processador barato, não possui ULA, garantindo que a execução fique próxima de 90%.

Por haver apenas um core, o processador divide todo seu tempo (100%) entre as diferentes Tasks, independente de quantas Threads cada uma possui.

Por haver vários cores, o processador permite a execução de várias Tasks sempre mantendo a taxa de utilização baixa.

Por haver apenas um core, o processador divide todo seu tempo (100%) entre as diferentes Threads, independente de quantas Tasks cada uma possui.

Pergunta 9

0,08 / 0,08 pts

{Assista o vídeo:

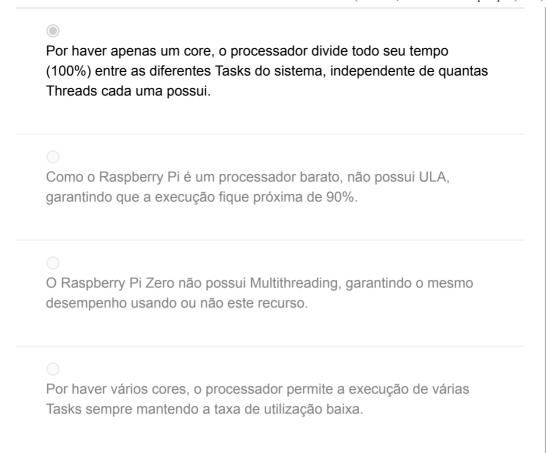
https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} No instante 5:00 o vídeo inicia um teste usando um Raspberry Pi Zero. A primeira Task disparou uma única Thread, consumindo 90% da CPU. Por que razão a segunda Task, que disparou múltiplas Threads, também consumiu 90% da CPU?

Por haver apenas um core, o processador divide todo seu tempo (100%) entre as diferentes Threads, independente de quantas Tasks cada uma possui.



Pergunta 10

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} A partir do instante 7:25 o vídeo inicia um teste usando um Raspberry Pi 3. Por que razão o primeiro programa de teste executou em metade do tempo?

	Por au	ie a	CPU	alcancou	uma	velocidade	superior	а	1	00	00	%
--	--------	------	-----	----------	-----	------------	----------	---	---	----	----	---

Apesar de possuir apenas um core, este computador possui o dobro de memória.

Apenas por que o processador é mais rápido, não porque possui múltiplos núcleos.

Uma vez que a tarefa possui múltiplas Threads, esta é executada mais rapidamente.

Como o processador possui dois núcleos, consegue executar o programa na metade do tempo.

Pergunta 11

0,08 / 0,08 pts

{Assista o vídeo:

https://youtu.be/Tn0u-IIBmtc (https://youtu.be/Tn0u-IIBmtc)



(https://youtu.be/Tn0u-IIBmtc)

e responda} A partir do instante 8:43 o vídeo inicia um teste usando o mesmo Raspberry Pi 3. Por que razão o segundo programa de teste consumiu quase 400% da CPU?

Por possuir múltiplas tarefas, estas são divididas entre os diferentes núcleos existentes.

A indicação de 400% mostra que a CPU está além do limite de execução, exigindo o dobro de tempo para completá-la.

Apesar de mostrar 400%, este valor é irreal, pois a CPU de único core não pode passar de 100%

Como este computador possui 4 núcleos, as diferentes Threads da mesma Task foram distribuída entre os diferentes núcleos.	
Porque foi requerido o dobro da capacidade de processamento disponível no computador, pois possui apenas dois núcleos.	

Pergunta 12	0,08 / 0,08 pts
{Assista o vídeo: https://youtu.be/Tn0u-IIBmtc	(https://youtu.be/Tn0u-IIBmtc)
, ,	09:20 o vídeo inicia um teste usando o ndo instâncias de uma mesma Task icleos) cada Task está usando?
O 3	
2	
4	
Não há como determinar qu	antos cores cada Task usa.
O 1	

Pontuação do teste: 1 de 1