

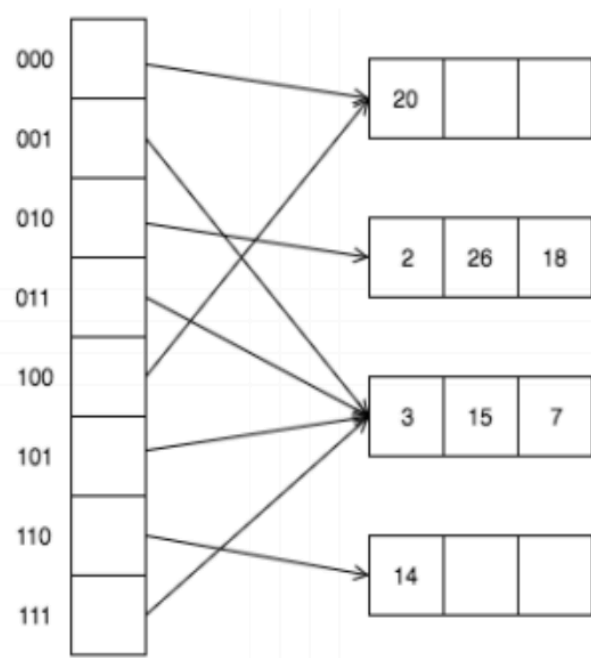
Quiz 05 - Hash

- Entrega 24 set em 21:00
- Pontos 4
- Perguntas 4
- Disponível 22 set em 0:00 - 24 set em 23:59
- Limite de tempo 20 Minutos

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	Tentativa 1	16 minutos	4 de 4
<div>❗ As respostas corretas estão ocultas.</div>			
Pontuação deste teste: 4 de 4			
Enviado 23 set em 13:39			
Esta tentativa levou 16 minutos.			
⋮			
Pergunta 1			
1 / 1 pts			
Qual é a vantagem do uso de cestos (<i>buckets</i>) nas tabelas <i>hash</i> em disco?			
<input type="radio"/> Permitir a criação de funções <i>hash</i> mais eficientes.			
<input type="radio"/> Criar uma tabela <i>hash</i> de tamanho variável.			
<input type="radio"/> Reduzir o tamanho da tabela e, assim, o espaço em disco ocupado por ela.			
<input checked="" type="radio"/> Aumentar a eficiência no armazenamento e na recuperação dos dados em disco.			
Os cestos (<i>buckets</i>) são estruturas que comportam um conjunto de elementos em um único endereço de uma tabela <i>hash</i> . Isso reduz, proporcionalmente, o número de colisões. Dessa forma, acelera o armazenamento de elementos, como também otimiza a leitura, uma vez que facilita a recuperação mais ágil dos elementos. É importante considerar que a leitura de um bucket ou de um único elemento geralmente envolve o mesmo custo de acesso a disco.			
⋮			
Pergunta 2			
1 / 1 pts			
Em uma tabela hash extensível, é possível afirmar que:			
<input type="radio"/> A cada vez que o diretório duplica seu tamanho, os <i>buckets</i> também são duplicados.			

<input checked="" type="radio"/>	Quando a tabela cresce, apenas as entradas no <i>bucket</i> dividido precisam ser reposicionadas.
<input type="radio"/>	Os <i>buckets</i> são acessados por meio de em uma lista encadeada de <i>buckets</i> .
<input type="radio"/>	A divisão de um <i>bucket</i> de profundidade menor que a profundidade do diretório não provoca qualquer mudança no diretório.
	A tabela <i>hash</i> extensível pode crescer ou diminuir de acordo com a necessidade. Quando um <i>bucket</i> é dividido, o diretório deve ser atualizado (duplicado, se necessário) e apenas as entradas desse <i>bucket</i> dividido precisam ser reposicionadas, pois podem ficar tanto no <i>bucket</i> antigo quanto no <i>bucket</i> recém-criado.
⋮	
Pergunta 3	
1 / 1 pts	
O que pode ser calculado a partir da profundidade do diretório de uma tabela <i>hash</i> extensível?	
<input checked="" type="radio"/>	A quantidade de endereços no diretório.
<input type="radio"/>	A quantidade de <i>buckets</i> presentes na tabela <i>hash</i> .
<input type="radio"/>	A quantidade de elementos inseridos na tabela <i>hash</i> .
<input type="radio"/>	A quantidade máxima de elementos em cada <i>bucket</i> .
O diretório de uma tabela <i>hash</i> é uma estrutura que aponta para cada <i>bucket</i> . É importante, portanto, que tenha uma quantidade maior ou igual de endereços que a quantidade de <i>buckets</i> existentes na tabela. Esses endereços (ponteiros para os <i>buckets</i>) são calculados a partir da profundidade do diretório. Assim, um diretório de profundidade 3 consegue apontar para até $2^3 = 8$ <i>buckets</i> . O número de <i>buckets</i> , porém, pode ser inferior ao número de endereços, pois o ponteiro de um <i>bucket</i> pode ser usado em mais de um endereço.	
⋮	
Pergunta 4	
1 / 1 pts	
Considere a seguinte tabela <i>hash</i> extensível:	



Suponha que façamos a inserção das chaves 30, 21, 12, 42 e 31, nesta ordem. Que chave provocará o primeiro aumento de profundidade do diretório?

Considerando a função *hash* $h(x) = x \bmod 2^p$, a chave 30 caberá no quarto *bucket*. A chave 21 será alocada para o terceiro *bucket*, provocando a sua divisão, mas esse *bucket* tem profundidade local menor que a profundidade global, então não promove a duplicação do diretório. A chave 12 ficará no primeiro *bucket*. A chave 42 será direcionada para o segundo *bucket*, que deve ser dividido. No entanto, como a sua profundidade local é igual à profundidade global, será necessário o aumento do diretório.

Pontuação do teste: 4 de 4