

이 파일에서는 메모리 빈 공간 관리에 대해 설명하고 있습니다. 가상 메모리 관리 시스템에서 중요한 주제로, 메모리 할당 과정에서 단편화 문제를 해결하는 방법들이 다루어집니다. 여기 요약과 함께 시험 대비를 위한 핵심 사항을 정리해드릴게요.

요약

1. 빈 공간 관리 문제:

고정 크기 공간은 관리가 쉬우나, 가변 크기 공간은 외부 단편화 문제가 발생할 수 있습니다.

외부 단편화란 빈 공간이 여러 작은 조각으로 나뉘어 실제로는 충분한 메모리가 있어도 연속된 공간이 없어 할당이 실패하는 문제를 말합니다.

2. 할당 및 해제:

malloc()은 원하는 크기의 메모리를 할당하고, free()는 메모리를 해제합니다.

할당기는 빈 공간을 추적하는 리스트를 관리하며, 빈 공간을 할당 및 해제할 때 병합(coalescing)과 분할(splitting) 기법을 사용합니다.

3. 기본 메모리 할당 전략:

최적 적합 (Best Fit): 요청에 맞는 가장 작은 빈 공간을 찾습니다. 단점은 리스트 전체를 검색해야 하므로 시간이 오래 걸릴 수 있습니다.

최악 적합 (Worst Fit): 가장 큰 빈 공간을 찾아 할당하고, 남은 공간은 다시 빈 공간으로 남깁니다. 큰 조각이 남아 단편화를 줄일 수 있지만, 성능이 저하될 수 있습니다.

최초 적합 (First Fit): 요청보다 큰 첫 번째 빈 공간을 할당합니다. 탐색 시간이 짧지만 작은 조각들이 많이 남을 수 있습니다.

다음 적합 (Next Fit): 이전에 할당된 지점부터 탐색을 시작해 할당합니다. 탐색이 고르게 분산됩니다.

4. 다른 기법:

슬랩 할당기 (Slab Allocator): 자주 요청되는 특정 크기의 메모리 청크를 위한 캐시를 유지해 단편화를 줄이는 방법입니다.

버디 할당기 (Buddy Allocator): 2의 거듭제곱 크기의 메모리 블록을 할당하고, 할당 해제 시 인접한 버디 블록을 병합하여 큰 블록을 만듭니다.

시험 대비 포인트

1. 단편화 문제: 외부 단편화와 내부 단편화의 차이를 명확히 이해하세요. 외부 단편화는 빈 공간의 분할로 인한 문제이고, 내부 단편화는 할당된 메모리보다 실제로 사용되는 메모리의 크기가 작을 때 발생하는 문제입니다.

2. 주요 할당 전략:

최적 적합은 메모리 효율성에 좋지만 검색 비용이 큼니다.

최악 적합은 큰 빈 공간을 남기려는 전략이지만 단편화가 발생할 수 있습니다.

최초 적합은 가장 빠르지만, 작은 빈 공간들이 리

스트 앞부분에 많이 남을 수 있습니다.

3. 기법의 장단점:

분할과 병합: 분할(splitting)과 병합(coalescing)의 개념을 확실히 이해하세요. 할당과 해제 시 어떻게 작동하는지 알고 있어야 합니다.

슬랩 할당기: 운영체제에서 자주 쓰이는 자료구조를 위한 캐시 개념을 알아두세요. 빠른 메모리 할당과 단편화 방지에 효과적입니다.

4. 실제 시스템에서의 적용:

버디 할당기: 할당과 해제 시 어떻게 버디 블록을 병합하고, 주소를 이용해 빠르게 병합할 수 있는

지 이해하세요.

학습 팁

빈 공간 할당 전략의 성능 차이와 그 영향을 비교 분석하는 문제가 나올 수 있으니 각 기법의 장단점을 꼼꼼히 정리해보세요.

메모리 할당 과정에서 단편화 문제를 해결하기 위한 병합(coalescing)과 분할(splitting) 기법이 어떻게 동작하는지 명확히 이해하는 것이 중요합니다.

필요한 부분에 대해 더 자세한 설명이나 추가적인 질문이 있다면 말씀해주세요!