

⑥ ファイルシステムの実装(2)

基礎OS 2015年度(1組)

問1 割り当ての単位

オペレーティングシステムがファイル用のエリアを割り当てる単位は以下のどれか。

- A. ディレクトリ
- B. セクタ
- C. レコード
- D. シリンダ
- E. トラック
- F. ブロック**

APはOSに対し、**レコード**単位でディスクアクセス要求を行う。OSは、ディスクアクセスの回数を削減するために、複数のレコードをまとめた**ブロック**単位でアクセスする。これを**ブロック**という。ブロックの最大長は通常2¹²バイトとされる(例えば、4096バイト=2¹²バイト)。ディスクを効率的に使用するためには、ブロックの最大長を単位としてファイルエリアの割り当てを行う必要がある。この割り当ての単位も**ブロック**と呼ぶ。

問2 連続割り当て

ファイルの領域を連続したブロックで割り当てる方式は、割り当て開放を繰り返すことにより、穴が小さくなり、有効に利用できなくなる。この問題を解決するために必要な操作は、以下のどれか。

- A. デフラグメンテーション**
- B. シーク操作
- C. 動的記憶割り当て
- D. ソフトウェア割込み
- E. 外部断片化
- F. ブロックング

外部断片化を解消するためには、穴が連続するように全体を移動する必要がある。
この操作を**デフラグメンテーション**という。
尚、内部断片化とは、固定長のブロック単位で割り当てられるため、末尾の端数が未使用となり無駄になることを言う。

【上記赤字の用語は、記憶領域の管理の重要語であり、理解しておくこと】

重要: 連続割り当ての特徴

- ・ 利点
 - **アクセス速度が速い**
 - ・ 直接アクセスが可能: 目的のレコード番号からブロック番号が計算できる
 - ・ 連続してアクセスする場合、ディスクのシークが不要
 - 管理用のオーバーヘッドが少ない
- ・ 問題点
 - ファイルの生成時にサイズの指定が必要、サイズ拡張時は指定が必要(予め大きなサイズを割り当てる無駄、ファイルスペース管理は利用者責任)
- ・ 動的記憶割り当て(実行中に記憶領域の割り当てを行う)の方法
 - ブロックの管理リストの中から、大きな穴を見つける
 - ・ 割り当てられていない連続したブロックを穴という
 - **外部断片化**(割り当て、開放の繰り返しにより、連続した大きな穴が、小さな穴に分割され、有効に利用できなくなる)が発生する
 - ・ 合計では要求を満たしても、連続していなければ割り当てできない
 - ・ **内部断片化**(割り当てブロックに無駄ができること)との違いに注意
 - **詰め直し**(断片の穴を1つの連続穴になるように全体を移動)が必要

問3 アクセス速度

連続割り当てのアクセス速度が速い理由は、以下のどれか(2つ選択)。

- A. 直接アクセスが可能**
- B. 内部断片化が無い
- C. 信頼性が高い
- D. 管理用のオーバーヘッドが小さい
- E. 動的記憶割り当てができる
- F. ディスクヘッドの移動が少ない**

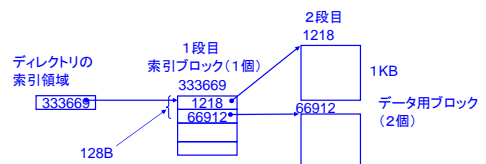
連続割り当ての場合、レコード番号からブロック番号が計算できるため、直接アクセスが可能(格納位置の探索が不要)であり、アクセス時間が短くなる。
また、ディスク上の連続域(同一トラック上のセクタ、同一シリンダ上のトラック)に記録されるため、ディスクヘッドを移動(シーク)せずに、アクセスが可能なので、アクセス時間が短い(シーク時間が0になる)。

問4 索引付き割り当てのオーバーヘッド

1ブロックの大きさが1024Bで、1152Bのファイルを1レベル索引付き割り当てで作成する場合、何ブロック必要か。尚、索引に必要な情報は、1索引当たり128Bとする。〔数値のみを半角数字で記入〕

答 3

データ用ブロック: $1152 \div 1024 = 1.5 \rightarrow 2$ ブロック } 合計3ブロック
索引用情報: $128B \times 2 = 256B \rightarrow 1$ ブロックで収容可能
真面目に計算するなら、索引ブロック1個当たりのエントリ数 = $1024 \div 128 = 8$
索引ブロック数 = $2 \div 8 = 0.25 \rightarrow 1$



OS2015⑥-1

割当て方式のオーバーヘッド比較

連続割当て: デリトリに先頭ブロック番号と長さ

33669 → 33669 33670 33671

33669 3

鎖状割当て: デリトリに先頭、最終ブロック番号
各データブロックに次ブロック番号

33669 → 15015 → 27693 → null

33669 15015 27693

索引付き割当て: デリトリに索引ブロック番号
索引ブロックにデータブロック番号

33669 → 15015 → 27693 → 44319

33669 15015 27693 44319

データブロックが1個でも
索引ブロック1個が必要

オーバーヘッド
連続割当て<鎖状割当て<索引付き割当て

OS2015⑥-1

問5 鎖状割当てのオーバーヘッド

1ブロックの大きさが1024Bの場合、2MBのファイルを鎖状割り当てで作成するために必要なブロック数を求めよ。但し、次ブロックへのポインタのために、8B必要なものとする。尚、1MB=1024KB、1KB=1024Bとする。

【数値のみを半角数字で記入】

答 2065

1ブロックのデータ部長
= ブロックサイズ - ポインタ部
= 1024 - 8 = 1016[バイト]

必要なブロック数 = ファイルサイズ / 1ブロックのデータ部の長さ
= 2 × 1024 × 1024 / 1016 = 2064.12 → 2065ブロック

データ部

ポインタ

1016n[バイト]

8バイト

データ部

ポインタ

OS2015⑥-1

問6 ファイル割当て

1ブロック2048Bのオペレーティングシステムが、2レベル索引によるファイル領域の割り当てを行った。この領域に格納できるファイルデータは最大で何MBか。尚、索引に必要な情報は、1索引当たり16Bとする。また、1KB=1024B、1MB=1024KB(1024=2¹⁰乗)とする。【数値のみを半角数字で記入】

32 [MB]

1つの索引ブロックで管理できる下位のブロック数は2048/16= 128=2⁷
2レベル索引なので、1つの索引ブロックの下に128個の索引ブロックができ、それぞれが128個のデータ用ブロックを管理できる。その数は128×128=2⁷×2⁷=2¹⁴。
データ領域のサイズは2048B×2¹⁴=2¹¹B×2¹⁴=2²⁵B=32×2¹⁰B=32MB
(2048B×128×128=33554432B=32768KB=32MB)

索引ブロック

索引ブロック

データ用ブロック

128個

128個

128個

128²個

OS2015⑥-1

問7 ファイル割当て

1ブロック16KBで磁気ディスクのデータ領域を割り当てるファイルシステムがある。このデータ領域一杯に、大きさが50KBのファイルと8KBのファイルを個数の比率1:1で作成した場合、磁気ディスクの全データ領域に占める実データの割合は約何%か。ここで、磁気ディスクの容量は十分に大きいものとする。【%値のみを半角数字で置き、解答欄に記入。必要なら小数点以下を四捨五入。】(初級シスアド 平成19年度春期間3改)

73 [%]

ファイルの保存に必要なブロック数=ファイルサイズ/ブロック長(切上げ)
50KBのファイルの保存は、50/16=3.125→4ブロック。
8KBのファイルの保存は、8/16=0.5→1ブロック。
この2種類のファイルが1:1の割合で、データ領域全体に格納される。即ち、5ブロック(16KB×5=80KB)ごとに、50KB+8KB=58KBの実データが入っている。したがって実データの割合は、58/80=0.725。

基礎OS⑥-1

スライド(問8の添付ファイル)

ビット番号

0 1 2 3 4 5 6 7 番地

0 1 1 1 1 1 1 1 2100

1 1 1 1 1 1 1 1 2101

2 1 1 1 1 1 1 1 2991

3 0 0 0 0 0 1 0 2992

4 0 0 1 1 1 0 1 2993

5 0 1 1 1 1 0 0 1 2994

6 0 0 0 0 0 0 1 0 2995

オール1

1KB=10¹⁰B=1024B

1MB=10²⁰B=1024KB

OS2015⑥-1

問8 ビットマップによる管理

2100番地以降に、スライド(添付ファイル)に示すようなビットマップによる空きブロックの管理エリアがある。尚、途中省略している部分は、オール1である。穴の数は幾つか。【数値のみを半角数字で記入】

答 7

ビット番号

0 1 2 3 4 5 6 7 番地

0 1 1 1 1 1 1 1 2100

1 1 1 1 1 1 1 1 2101

2 1 1 1 1 1 1 1 2991

3 0 0 0 0 0 1 0 2992

4 0 0 1 1 1 0 1 2993

5 0 1 1 1 1 0 0 1 2994

6 0 0 0 0 0 0 1 0 2995

0: 空き

1: 使用中

穴: 空きブロックの連続(1個も含めて連続した0)

③の続き

④の続き

問9 ビットマップによる管理

OS2015⑧-1

スライド【図8の添付ファイル】の管理エリアにおいて、ブロックサイズが2KBの場合、管理可能なファイル領域全体の大きさは何MBか。但し、番地は10進数、1KB、1MBは、スライドの値とする。【数値のみを半角数字で記入】

答 14

ビット番号

管理エリアのバイト数
=2995+1-2100=896
(1を足さないで2100番地を含めないことになる。管理エリアが、2100番地の1バイトだけの場合を考えてみよ。)

管理できるブロック数
=896÷8=7168 (=7×1024)
ファイル領域サイズ
=2KB×7168
=2×1024×7×1024B
=14MB

バイト番号

	0	1	2	3	4	5	6	7	番地
0	1	1	1	1	1	1	1	1	2100
1	1	1	1	1	1	1	1	1	2101
:	:	:	:	:	:	:	:	:	:
	1	1	1	1	1	1	1	1	2991
	0	0	0	0	0	1	0	0	2992
	0	0	1	1	1	0	1	0	2993
	0	1	1	1	1	0	0	1	2994
	0	0	0	0	0	0	1	0	2995

問10 ビットマップによる管理

OS2015⑧-1

スライド【図8の添付ファイル】の管理エリアにおいて、ブロックサイズ3のファイル用のエリアが連続割当てで要求された。要求を満たす最小の穴のブロックを若番から割当てする場合、先頭のブロック番号は幾つか。【数値のみを半角数字で記入】

答 7142

ビット番号

割当て先頭
2992番地オフセット6
バイト番号k=2992-2100=892
(語数の場合と異なり、語番号の場合は2100番地が0番なので、2100を減算するだけで良い)

ブロック番号
i=8×892+6=7142

バイト番号

	0	1	2	3	4	5	6	7	番地
0	1	1	1	1	1	1	1	1	2100
1	1	1	1	1	1	1	1	1	2101
:	:	:	:	:	:	:	:	:	:
891	1	1	1	1	1	1	1	1	2991
892	0	0	0	0	0	1	0	0	2992
893	0	0	1	1	1	0	1	0	2993
894	0	1	1	1	1	0	0	1	2994
895	0	0	0	0	0	0	1	0	2995

緑色のブロックが割り当てられる。