

第4章 ファイルシステム 演習問題

1. ファイル `/etc/passwd` に対する5つの異なるパス名を挙げなさい。(ヒント: ディレクトリのエントリ「`.`」と「`..`」について考えなさい。)
2. Windowsでは、ユーザーがWindowsエクスプローラにリストされたファイルをダブルクリックすると、プログラムが実行され、そのファイルがパラメータとして渡されます。オペレーティングシステムがどのプログラムを実行すべきかを知るための2つの異なる方法を挙げなさい。
3. 初期のUNIXシステムでは、実行可能ファイル(a.outファイル)はランダムに選ばれたものではなく、非常に特定のマジックナンバーで始まっていました。これらのファイルはヘッダで始まり、その後にテキストセグメントとデータセグメントが続きました。なぜ実行可能ファイルには非常に特定の番号が選ばれ、他のファイルタイプには多かれ少なかれランダムなマジックナンバーが先頭ワードとして使われたのだと思いますか。
4. UNIXのopenシステムコールは絶対に不可欠ですか。それが無い場合、どのような結果になりますか。
5. シーケンシャルファイルをサポートするシステムには、常にファイルを巻き戻す操作があります。ランダムアクセスファイルをサポートするシステムも、これを必要としますか。
6. 一部のオペレーティングシステムは、ファイルに新しい名前を付けるためのrenameシステムコールを提供しています。このコールを使ってファイルの名前を変更することと、単に新しい名前で作成した新しいファイルにコピーし、古いファイルを削除することとの間に何か違いはありますか。
7. 一部のシステムでは、ファイルの一部をメモリにマップすることが可能です。そのようなシステムはどのような制限を課さなければなりませんか。この部分的なマッピングはどのように実装されますか。
8. ある単純なオペレーティングシステムは単一のディレクトリしかサポートしませんが、任意に長いファイル名を持つファイルを任意に多数持つことを許可します。階層的なファイルシステムに近似するものをシミュレートできますか。どのようにしますか。
9. UNIXとWindowsでは、ランダムアクセスは、ファイルに関連付けられた「現在位置」ポインタをファイル内の指定されたバイトに移動させる特別なシステムコールを持つことによって行われます。このシステムコールなしでランダムアクセスを行う代替方法を提案しなさい。
10. 図4-8のディレクトリツリーを考えます。もし `/usr/jim` がワーキングディレクトリである場合、相対パス名が `../ast/x` であるファイルの絶対パス名は何ですか。
11. 本文で述べたように、ファイルの連続割り当てはディスクのフラグメンテーションを引き起こします。なぜなら、ファイルの長さがブロックの整数倍でない場合、最後のディスクブロックに無駄なスペースが生じるからです。これは内部フラグメンテーションですか、それとも外部フラグメンテーションですか。前の章で議論した何かと類推しなさい。
12. 与えられたファイルに対する破損したデータブロックの影響を、(a) 連続割り当て、(b) リンク割り当て、および (c) インデックス (またはテーブルベース) 割り当てについて説明しなさい。
13. ディスクの連続割り当てを使用し、ホールに悩まされない一つの方法は、ファイルが削除されるたびにディスクをコンパクションすることです。すべてのファイルが連続しているため、ファイルのコピーにはシークと回転遅延でファイルを読み取り、その後フルスピードで転送する必要があります。ファイルを書き戻すにも同じ作業が必要です。シーク時間が5ミリ秒、回転遅延が4ミリ秒、転送レートが80MB/秒、平均ファイルサイズが8KBであると仮定した場合、ファイルをメインメモリに読み込み、それをディスクの新しい場所に書き戻すのにどれくらいの時間がかかりますか。これらの数値を使用して、16GBのディスクの半分をコンパクションするのにどれくらいの時間がかかりますか。
14. 前の問題の答えに照らして、ディスクのコンパクションは意味がありますか。
15. 一部のデジタル家電は、例えばファイルとしてデータを保存する必要があります。ファイルの保存が必要で、連続割り当てが優れたアイデアとなる現代のデバイスを挙げなさい。
16. 図4-13に示されているiノードを考えます。もしそれが10個のダイレクトアドレスを含み、これらがそれぞれ8バイトで、すべてのディスクブロックが1024KBだった場合、可能な最大のファイルは何ですか。
17. あるクラスについて、学生の記録がファイルに保存されています。記録はランダムにアクセスされ、更新されます。各学生の記録が固定サイズであると仮定します。3つの割り当て方式(連続、リンク、テーブル/インデックス)のうち、どれが最も適切ですか。
18. その寿命の間にサイズが4KBから4MBの間で変動するファイルを考えます。3つの割り当て方式(連続、リン

ク、テーブル/インデックス)のうち、どれが最も適切ですか。

19. iノード内に短いファイルのデータを保存することで、効率を改善しディスクスペースを節約できると提案されています。図4-13のiノードの場合、iノード内に何バイトのデータを保存できますか。
20. 2人のコンピュータ科学の学生、キャロラインとエリナーがiノードについて議論しています。キャロラインは、メモリが非常に大きく安価になったため、ファイルが開かれるときに、iノードテーブル全体を検索して既に存在するかどうかを確認するよりも、新しいiノードのコピーをiノードテーブルにフェッチする方が単純で速いと主張しています。エリナーは反対です。どちらが正しいですか。
21. ハードリンクがシンボリックリンクより優れている点を1つ、シンボリックリンクがハードリンクより優れている点を1つ挙げなさい。
22. ハードリンクとソフトリンクがiノードの割り当てに関してどのように異なるか説明しなさい。
23. 4KBのブロックとフリーリスト方式を使用する4TBのディスクを考えます。1つのブロックにいくつのブロックアドレスを保存できますか。
24. 空きディスクスペースは、フリーリストまたはビットマップを使用して追跡できます。ディスクアドレスにはDビットが必要です。Bブロックのディスクで、そのうちFが空きである場合、フリーリストがビットマップより少ないスペースを使用する条件を述べなさい。Dが16ビットの値を持つ場合、あなたの答えを空きでなければならないディスクスペースのパーセンテージとして表現しなさい。
25. ディスクパーティションが最初にフォーマットされた後の空き領域ビットマップの始まりは次のようになっています: 1000 0000 0000 0000 (最初のブロックはルートディレクトリによって使用されます)。システムは常に最も低い番号のブロックから空きブロックを検索するので、6ブロックを使用するファイルAを書き込んだ後、ビットマップは次のようになります: 1111 1110 0000 0000。以下の追加アクションのそれぞれの後のビットマップを示しなさい: (a) 5ブロックを使用するファイルBが書き込まれる。(b) ファイルAが削除される。(c) 8ブロックを使用するファイルCが書き込まれる。(d) ファイルBが削除される。
26. 空きディスクブロックに関する情報を含むビットマップまたはフリーリストがクラッシュにより完全に失われた場合、何が起こりますか。この災害から回復する方法はありますか、それともディスクはさようならですか。UNIXとFAT-16ファイルシステムについて、あなたの答えを別々に議論しなさい。
27. オリバー・オウルの大学計算センターでの夜の仕事は、夜間のデータバックアップに使用されるテープを交換することです。各テープが完了するのを待つ間、彼はシェイクスピアの戯曲が地球外訪問者によって書かれたことを証明する論文を書いています。彼のテキストプロセッサは、彼らが持っている唯一のシステムであるバックアップ中のシステムで実行されます。この配置に問題はありますか。
28. 本文でインクリメンタルダンプについて詳しく議論しました。Windowsでは、すべてのファイルにアーカイブビットがあるため、いつダンプするかを判断するのは簡単です。このビットはUNIXにはありません。UNIXのバックアッププログラムはどのファイルをダンプすべきかをどのように知りますか。
29. 図4-25のファイル21が最後のダンプ以降変更されていないと仮定します。図4-26の4つのビットマップはどのように異なりますか。
30. 各UNIXファイルの最初の部分をそのiノードと同じディスクブロックに保持することが提案されています。これにはどのような利点がありますか。
31. 図4-27を考えます。特定のブロック番号に対して、両方のリストのカウンタが2の値を持つことは可能ですか。この問題はどのように修正すべきですか。
32. ファイルシステムの性能はキャッシュのヒット率(キャッシュで見つかるブロックの割合)に依存します。キャッシュからのリクエストを満たすのに1ミリ秒かかり、ディスク読み取りが必要な場合は40ミリ秒かかるとすると、ヒット率がhの場合にリクエストを満たすのに必要な平均時間の式を求めなさい。この関数をhが0から1.0まで変化する値に対してプロットしなさい。
33. コンピュータに接続された外部USBハードドライブには、ライトスルーキャッシュとブロックキャッシュのどちらがより適していますか。
34. 学生の記録がファイルに保存されているアプリケーションを考えます。アプリケーションは学生IDを入力として受け取り、その後対応する学生の記録を読み取り、更新し、書き込みます。これはアプリケーションが終了するまで繰り返されます。「ブロック先行読み込み(block read-ahead)」技術はここで役立ちますか。
35. ブロック14から23までの10個のデータブロックを持つディスクを考えます。ディスク上にはf1とf2の2つのファイルがあります。ディレクトリ構造には、f1とf2の最初のデータブロックがそれぞれ22と16であると記載されています。以下のFATテーブルエントリが与えられた場合、f1とf2に割り当てられているデータブロックは何で

- すか。(14,18); (15,17); (16,23); (17,21); (18,20); (19,15); (20, -1); (21, -1); (22,19); (23,14) 上記の表記では、(x, y)はテーブルエントリxに格納されている値がデータブロックyを指すことを示します。
36. 図4-21の背後にある考えを考えますが、今度は平均シーク時間が6ミリ秒、回転数が15,000rpm、トラックあたり1,048,576バイトのディスクについてです。ブロックサイズがそれぞれ1KB、2KB、4KBの場合のデータレートは何ですか。
37. あるファイルシステムは4KBのディスクブロックを使用しています。ファイルサイズの中央値は1KBです。すべてのファイルが正確に1KBだった場合、ディスクスペースの何パーセントが無駄になりますか。実際のファイルシステムでの無駄は、この数値よりも高くなると思いますか、それとも低くなると思いますか。あなたの答えを説明しなさい。
38. 4KBのディスクブロックサイズと4バイトのブロックポインタアドレス値が与えられた場合、10個のダイレクトアドレスと1個の間接ブロックを使用してアクセスできる最大のファイルサイズ（バイト単位）は何ですか。
39. MS-DOSのファイルは、メモリ内のFAT-16テーブルのスペースを競合しなければなりません。もし1つのファイルがk個のエントリを使用する場合、それは他のどのファイルも利用できないk個のエントリです。これは、すべてのファイルの合計長にどのような制約を課しますか。
40. あるUNIXファイルシステムは4KBのブロックと4バイトのディスクアドレスを持っています。iノードがそれぞれ10個のダイレクトエントリ、1個のシングル、ダブル、トリプル間接エントリを含む場合、最大のファイルサイズは何ですか。
41. パス名 /usr/ast/courses/os/handout.t を持つファイルのiノードを取得するために、いくつかのディスク操作が必要ですか。ルートディレクトリのiノードは常にメモリにあると仮定し、パス上の他のものはメモリにないとします。また、すべてのディレクトリは1つのディスクブロックに収まると仮定します。
42. 多くのUNIXシステムでは、iノードはディスクの先頭に保持されます。代替の設計は、ファイルが作成されるときにiノードを割り当て、iノードをファイルの最初のブロックの先頭に置くことです。この代替案の長所と短所を議論しなさい。
43. ファイルのバイトを逆順にするプログラムを書きなさい。最後のバイトが最初になり、最初のバイトが最後になるようにします。これは任意に長いファイルで動作しなければなりません、合理的に効率的にするように試みなさい。
44. 与えられたディレクトリから始まり、その時点からファイルツリーを降りていき、見つけたすべてのファイルのサイズを記録するプログラムを書きなさい。すべてが終わったら、パラメータとして指定されたビン幅を使用してファイルサイズのヒストグラムを印刷すべきです（例：1024の場合、0から1023のファイルサイズは1つのビンに、1024から2047は次のビンに入ります）。
45. UNIXファイルシステム内のすべてのディレクトリをスキャンし、ハードリンクカウントが2以上のすべてのiノードを見つけて特定するプログラムを書きなさい。そのようなファイルごとに、そのファイルを指すすべてのファイル名をまとめてリストアップしなさい。
46. UNIXのlsプログラムの新しいバージョンを書きなさい。このバージョンは、引数として1つ以上のディレクトリ名を取り、各ディレクトリについてそのディレクトリ内のすべてのファイルを1行に1ファイルずつリストアップします。各フィールドは、そのタイプに応じて合理的な方法でフォーマットされるべきです。ディスクアドレスがある場合は、最初のディスクアドレスのみをリストアップしなさい。
47. アプリケーションレベルのバッファサイズが読み取り時間に与える影響を測定するプログラムを実装しなさい。これには、大きなファイル（たとえば2GB）への書き込みと読み取りが含まれます。アプリケーションのバッファサイズを（たとえば64バイトから4KBまで）変更します。タイミング測定ルーチン（UNIXのgettimeofdayやgetitimerなど）を使用して、異なるバッファサイズでの所要時間を測定します。結果を分析し、あなたの発見を報告しなさい：バッファサイズは全体の書き込み時間と書き込みごとの時間に違いをもたらしますか。
48. ディスクに保存された単一の通常ファイルに完全に含まれるシミュレートされたファイルシステムを実装しなさい。このディスクファイルには、ディレクトリ、iノード、空きブロック情報、ファイルデータブロックなどが含まれます。空きブロック情報を維持し、データブロックを割り当てるための適切なアルゴリズム（連続、インデックス、リンク）を選択しなさい。あなたのプログラムは、ユーザーからのシステムコマンドを受け付けて、ディレクトリの作成/削除、ファイルの作成/削除/オープン、選択されたファイルからの読み取り/書き込み、ディレクトリ内容のリスト表示を行います。