

1) 次の文の空欄を埋めよ。(0.2×12=2.4点)

デッドロックは(.....)が(.....)している/なっている状態のことで、原因は(.....)が(.....)していることによる。プロセス A と B がデッドロックする件は、①②つの(.....)的な資源を同時に必要としており、②資源確保時に(.....)(する)方式で、③資源の横取りが不可であり、④(.....)になっていること、が条件になる。4つの条件のうち1つでも欠けると、デッドロックにならない。

デッドロックを起こしたときは、プロセスは(.....)の状態にあり、処理は(.....)。しかしシステム(OS カーネル)から見るといずれのプロセスも(.....)な状態であり、他のプロセスと区別できない。更に、同じプログラムであっても、デッドロックを起こすかどうかは(.....)に依存するので、プログラムの処理手順を見ても起こすかどうかは(.....)。

2) デッドロックの対応について次の間に答えよ。(0.8×2=1.6点)

(あ) デッドロックが起きてしまったから、その状態を検出して解決する対処法が考えられる。具体的に、どうやって検出するか、検出されたときにどうやって解決するか、説明せよ。

(検出の方法)

(解決の方法)

(い) 初めからプログラムを、デッドロックが絶対に起こらないように考えて設計しておけばよいと言えるだろう。絶対に起こらないようにするためには、どのような手法が考えられるか、説明せよ。

3) (時間があれば) デッドロックについて、なぜそんなに騒ぐのかを考えてみよう。(点なし)

a) まず、並行(並列)処理で資源を取り合わなければよいだろう。資源を取り合うような処理が、どうして(どういう場合に)必要なのか?

b) では、並行+資源を取り合う(排他資源を共有する)ことは必要だとして、上記2)の(い)で考えたデッドロックの対応法を使おうと思う。できればデッドロックが起こらないようにあらかじめプログラムを作っておきたい。(起こってから対応するのは無駄が多いから)

起こらないようにする方法は、確かにデッドロックは起こさなくなるが、その代償を払わなければならない。それは何だろうか?

4) ハードディスクのエレベータ式(教科書では SCAN 方式/LOCK 方式)のスケジューリングについて、

a) どういうやり方が説明せよ。(0.5 点)

.....

.....

.....

b) 表のアクセス待ちがあり、スタート点を 50、最初の方向を下向き(番号が小さくなる方向)とするとき、エレベータ式に動かすとどういふ順番になるか?(0.5 点)

待ちの順番	始め	1	2	3	4	5	6	7	8	9	10
アクセス場所	50	34	7	85	43	61	79	54	95	12	23

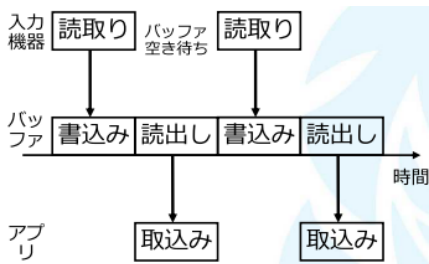
並べ替えた後の場所	50	43									
-----------	----	----	--	--	--	--	--	--	--	--	--

(時間があれば) 移動距離の合計を計算して、エレベータの方が距離が小さいことを確認せよ。(点なし)

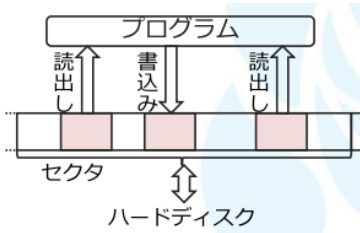
.....

.....

5) バッファリングによる効率向上の考え方を、(下の図を使って)説明せよ (1点)



6) ブロッキングによる効率向上の考え方を、(図を使って)説明せよ (1点)



7) プリフェッチ(先読み)の考え方を説明せよ (1点)

.....

.....

.....

.....

.....