オペレーティングシステム・期末試験の解答と解説

2014 年度 E・O クラス (2015 年 2 月 9 日・試験時間 90 分)

1.(a) 図 1 の通り.

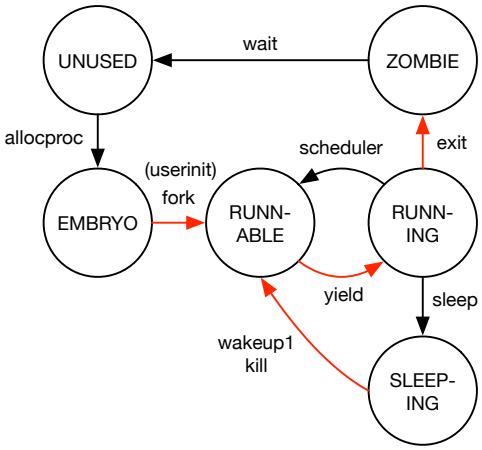

図 1: xv6 のプロセスの状態（解答）

補足 各遷移のラベルはその遷移を起こす関数名である．これは解答に書く必要はないが参考のために掲載する.

(b)

UNUSED	8	EMBRYO	2
RUNNABLE	1	RUNNING	7
SLEEPING	6	ZOMBIE	4

(c) n

(d) プリエンプティブ方式

補足 「プリエンプティブ」,「横取り（方式）」,「プリエンプティブスケジューリング」,「横取りスケジューリング」でもよい.

(e) `wait`

(f) (3)

補足 問題文にミスがあって選択肢 (3) が 2 個あったが，正解も (3) なので特に修正なしとする.

解説 2 番目の選択肢 (シリンダーロック) は昔から使われている錠前の一種．(1) は音楽ジャンルの一つ（プログレッシブロックと呼ばれるジャンルの一種）で排他制御とは何の関係もない.

(g) ロックする期間が短い場合，あるいはカーネル内スレッドでのロックを実現したい場合

補足 どちらか一方のみの解答でもよい.

解説 スリープロックを行うためにはプロセス（スレッド）を sleeping(waiting) 状態にする必要がある．そのために一旦カーネルモードを経由することになり，そのオーバーヘッドは無視できない．スピンロックは CPU を無駄に消費しているとも考えられるが，ロックされる期間が短い場合は相対的なオーバーヘッドは小さくなる.

また，スリープロックを行う際にはスレッド（プロセス）スケジューリングの機構が必要であるが，カーネル内スレッドではその機構を用いることができない場合がある．そのような場合はスピンロックを用いることになる．xv6 のカーネルスレッドも xchg 命令によるスピンロックを行っている.

(h) 当該プロセスの（`proc` 構造体の）初期化中に，別の CPU による初期化やスケジュールを防止するため.

解説 xv6 では，配列 `ptable.proc` の要素（`proc` 構造体）として PCB を管理している．新しいプロセスを起動する際には，`ptable.proc` の要素（`proc` 構造体）でフィールド `state` の値が `UNUSED` であるものを

探し，その構造体を初期化する．その際，state の値が UNUSED のままだと，他の CPU によって初期化が開始される恐れがある．また，state の値を RUNNABLE にしてしまうと，他の CPU のスケジューラによって（すでに初期化が終わったとみなされて）RUNNING にされてしまう可能性がある．つまり，UNUSED でも RUNNABLE でもない，初期化作業中であることを示す状態が必要となる．

2. (a) 71680 バイト（70K バイト）

解説 直接参照 12 ブロックに加え，間接参照ブロックから $512 \div 4 = 128$ ブロックが参照できる．よって合計 $(12 + 128) \times 512 = 71680$ バイトまでのファイルを作ることができる．

(b) 19

解説 $\lceil 9000/512 \rceil = 18$ なのでデータブロックは 18 個必要である．加えて間接参照ブロックを必要とするため（直接参照できるのは 12 ブロックまでなので），合計 19 ブロックとなる．

(c) 512

(d) ブートブロックを読み書きしてしまう恐れがある．

(e) ファイルへの書き込みを行う際に当該ブロックに上書きされてしまうおそれがある．

(f) ファイルやディレクトリを消去してもその inode 構造体（およびそこから参照されるデータブロック）がディスク上に残ってしまう．

(g) 4479

解説 ディレクトリに対するリンクは，その親ディレクトリからのものと，各子ディレクトリからのもの（“..” によるリンク）のみである．また，問題にあるようにディレクトリは link システムコールの第 1 引数とできないため，これによってディレクトリへのリンクが変更されることはない．よって，あるディレクトリの nlink の値は，その子ディレクトリの数を k とした場合に $k + 1$ となる．

次に，xv6 のファイルシステムにおいて，あるディレクトリの子ディレクトリをいくつ作れるかを考える．問題 **(a)** の解答にあるように最大のファイルサイズは 71680 バイトである．一方 dirent 構造体の大きさは 16 バイトなので，$71680 \div 16 = 4480$ 個のエントリを作ることができる．そのうち 2 個は “.” および “..” に用いられるので，子ディレクトリの最大数は 4478．よってあるディレクトリの nlink の最大値は 4479 となる．

(h) “..” のリンク先が定まらなくなる，子や孫から親へのループができることでファイルシステムの走査がしづらくなる等の不具合が生じる．

解説 ファイルシステムにおいて，ディレクトリ間の接続関係が木にならなくなる．つまり親ディレクトリを 2 つ以上持つディレクトリが存在し得ることになり，その結果として上記のような不具合が生じる．