

資料の持ち込み不可。電卓・電子辞書などの電子機器の利用不可。

1. 次の各記述について、最も適切であるものを答えよ。

- (1) 正しい記述はどれか。
 - (a) キーボードはブロックデバイスではなくキャラクタデバイスである。
 - (b) ゾンビプロセスに実行継続のシグナルを送ると、ゾンビプロセスが実行していたアプリケーションの実行を再開することができる。
 - (c) Linuxはマイクロカーネルの構造を持つOSである。
 - (d) コアが1つのCPUが1つしかない計算機では、並列実行は実現できるが、並行実行は実現できない。
- (2) UNIX系OSにおいてプロセスを生成するためのシステムコールはどれか。
 - (a) exec (b) CreateProcess (c) fork (d) creat
- (3) UNIX系OS上で、一つのプロセス内に複数のスレッドが走っているとする。各スレッド専用のものが使用されるのはどれか（他のスレッドと共有しないものはどれか）。なお、スレッドの種類は、UNIX系OSを中心に標準的に使われているPOSIXスレッド（Pthread）であると仮定せよ。
 - (a) 各レジスタの値 (b) 各メモリ領域の内容
 - (c) 割り込みベクタ (d) ファイルデスクリプタ
- (4) IA-32プロセッサ上で動作するLinuxにおけるページングのアドレス変換を考える。アプリケーションプログラム内の各機械語命令の各実行において、命令のオペランドに与えられた仮想アドレスから物理アドレスへの変換処理を行うのはどれか。
 - (a) デバイスドライバ (b) OSカーネル
 - (c) MMU (d) 標準ライブラリ
- (5) 正しい記述はどれか。
 - (a) TLB（Translation Lookaside Buffer）の目的は、ファイルパスをページ番号に変換する処理の高速化である。
 - (b) デマンド・ページングを行うOS上で動くプロセスが確保した仮想ページの中には、物理ページに対応付けられているものも、対応づけられていないものもある。
 - (c) 一つのOS上で動いている複数のプロセスは、同一のページテーブルを参照している。
 - (d) 複数の物理ページを一つの仮想ページに対応づけることにより、共有メモリを実現することができる。
- (6) スラッシングの問題を軽減するための対策として効果が期待できないものはどれか。
 - (a) メモリを増設する (b) ディスクを増設する
 - (c) プロセスの数を減らす (d) メモリ参照の局所性を上げる
- (7) 一つのディスクにおける次のもののうち、最も多くのバイト数を含むのはどれか。
 - (a) 1シリンダ (b) 1トラック (c) 1セクタ
- (8) 信頼性または安全性の向上に直接関係ないものはどれか。
 - (a) ジャーナリングファイルシステム (b) RAID
 - (c) 多段ページテーブル (d) IDS

- (9) 管理者が定めたセキュリティポリシーをプロセスの実行に対して強制的に適用するアクセス制御方式を何と呼ぶか。
(a) DAC (b) FAC (c) MAC (d) PAC
- (10) 割り込みに関する記述として誤っているものはどれか。計算機としてはIA-32プロセッサを搭載した最近のPCを仮定してよい。
(a) 割り込みが発生した際には、割り込み発生時点でのプログラムカウンタをCPUがメモリに保存する。
(b) PICやAPICなどの割り込みコントローラは、割り込みの調停を行うためのソフトウェアである。
(c) 割り込みマスクの仕組みにより、CPUは一部の割り込みを受け付けない状態にすることができる。
(d) ゼロで除算する命令を実行すると、割り込みが発生する。

2. 次の各用語を答えよ。

- (1) デバイス操作の命令ではなく、メモリの読み書き命令によって実現されるデバイス操作方式を何と呼ぶか。
(2) CPUを介さずデバイスとメモリの間で直接データを転送する方式を何と呼ぶか。
(3) ロックなどを用いて排他的に実行することが必要なプログラム部分を何と呼ぶか。
(4) 優先度の高いプロセスAが資源を占有し続け、プロセスBがその資源を必要としているにもかかわらず、プロセスBにその資源が永久に割り当てられないとする。プロセスBが陥っている状態を何と呼ぶか。
(5) 一度読み込んだファイル内容を将来の再度のアクセスに備えて保持するためにメモリ上に作られ、ファイルアクセスの高速化などに貢献するものを何と呼ぶか。

3. 以下は、UNIX系OS上で標準ライブラリのread関数を呼び出したときに行われる処理を説明したものであるが、順番が正しくない。正しい順に並べ替えよ。

- (a) 標準ライブラリのread関数の呼び出し
(b) プロセスの状態を実行可能状態に切り替え
(c) ユーザーモードに遷移し、システムコール割り込みを発行した命令の次の命令から実行を再開
(d) readシステムコールハンドラ（システムコールサービスルーチン）の呼び出し
(e) システムコール処理用割り込みハンドラの呼び出し
(f) プロセスの状態を待ち状態に切り替え
(g) 「データ到着」を意味するストレージデバイスからの割り込みがCPUに到着
(h) システムコール割り込みを発行する命令の実行によりカーネルモードに遷移
(i) ストレージデバイスの制御レジスタへの「データ読み出しコマンド」のセット
(j) 標準ライブラリのread関数からの復帰

4. 以下は擬似的なC言語のコードで書かれた、有限バッファを実現するプログラムの一部である。同期処理はロックと条件変数を操作するための以下の関数群によって実現されている。

- ・ `acquire_lock(lock_t *lk)`
lkが指すロックを獲得する。
- ・ `release_lock(lock_t *lk)`
lkが指すロックを解放する。
- ・ `cond_wait(condvar_t *cv, lock_t *lk)`
lkが指すロックの解放とcvが指す条件変数での待機を一息で行う。
本関数からの復帰前に、lkが指すロックを再び獲得する。
- ・ `cond_signal(condvar_t *cv);`
cvが指す条件変数で待機しているスレッドを再開させる。

(1)～(8)に入るべきコードを(a)～(f)から選んで答えよ。

```

=====
int buf[N]; int inp = 0, outp = 0; int count = 0;
lock_t lk;
condvar_t full; /* バッファに空きができる事象を待つための条件変数 */
condvar_t empty; /* バッファにデータが入る事象を待つための条件変数 */

/* Producerは以下の動作を繰り返す
 * ・ Produceを呼び出してデータを生産する
 * ・ バッファに空きがあればそのデータをバッファに入れる
 * ・ 空きがなければ待つ
 */
Producer()
{
    int data;
    for (;;) {
        data = Produce();
        (1)
        while (count == N) {
            (2)
        }
        count++;
        buf[inp] = data;
        inp = (inp + 1) % N;
        (3)
        (4)
    }
}

/* Consumerは以下の動作を繰り返す
 * ・ バッファにデータがあれば取り出す
 * ・ データがなければ待つ
 * ・ 取り出したデータをConsumeに渡して消費する
 */
Consumer()
{
    int data;
    for (;;) {
        (5)
        while (count == 0) {
            (6)
        }
        count--;
        data = buf[outp];
        outp = (outp + 1) % N;
        (7)
        (8)
        Consume(data)
    }
}
=====

```

- (a) acquire_lock(&lk);
- (b) release_lock(&lk);
- (c) cond_wait(&empty, &lk);
- (d) cond_wait(&full, &lk);
- (e) cond_signal(&empty);
- (f) cond_signal(&full);

5. シリンダ数が1000個（0番から999番）あるハードディスクを考える。現在ヘッドは300番シリンダの上にあり、アクセス要求を格納するキューには、

81, 912, 355, 721, 95, 578, 260

が入っている。各数字はシリンダ番号であり、要素が左にあればあるほど早くキューに挿入されたことを意味する。以降は要素がキューに挿入されることはない。ヘッドが隣のシリンダに移動するのに、1の時間がかかるとする。たとえば現在300番シリンダにあるヘッドが800番シリンダに移動するのに500の時間がかかる。要求を処理するための時間は0とする。ヘッドの移動に以下の各アルゴリズムを用いたとき、キュー内の全要求を処理し終わるまでの時間（すなわちヘッドの累積移動距離）を求めよ。

- (1) FCFS (First Come First Served)
- (2) SSTF (Shortest Seek Time First)
- (3) SCAN (ただし現在ヘッドはシリンダ番号が減る方向に動いているとする)

6. 次の各用語を1行から3行程度で簡潔に説明せよ。

- (1) iノード (i-node)
- (2) ページアウト (page out)
- (3) 特権命令 (privileged instruction)
- (4) DoS攻撃 (Denial-of-Service attack)
- (5) preemptive scheduling

7. 次の各問いに答えよ。

- (1) ページサイズが非常に大きい場合に発生しうる問題を一つ挙げよ。
- (2) ページサイズが非常に小さい場合に発生しうる問題を一つ挙げよ。
- (3) プロセスに割り当てる time quantum (time slice) が非常に大きい場合に発生しうる問題を一つ挙げよ。
- (4) プロセスに割り当てる time quantum (time slice) が非常に小さい場合に発生しうる問題を一つ挙げよ。
- (5) プロセスと比較したときの、スレッドの利点を一つ挙げよ。
- (6) カーネルレベルスレッドと比較したときの、ユーザレベルスレッドの利点を一つ挙げよ。
- (7) ユーザレベルスレッドと比較したときの、カーネルレベルスレッドの利点を一つ挙げよ。

8. 本科目（講義、ボーナス課題、期末試験など）に対する意見・感想があれば書いてください。来年度に参考にします。この部分は成績評価にまったく関係ありません。