

## 問010041解説

### ◆解答

設問 1 イ

設問 2 a ア b オ

設問 3 ウ

### ◆解説

排他制御に関する問題である。

P 1、P 2の演算処理に対して、次の4ケースが考えられる。

- ① P 1、P 2共に5を読み込み、演算後、P 1、P 2の順に書き込む。結果は4となる。
  - ② P 1、P 2共に5を読み込み、演算後、P 2、P 1の順に書き込む。結果は7となる。
  - ③ P 1が5を読み込み、演算後7を書込み、P 2が7を読み込み、演算後6を書き込む。  
結果は6となる。
  - ④ P 2が5を読み込み、演算後4を書込み、P 1が4を読み込み、演算後6を書き込む。  
結果は6となる。
- ①～④の演算結果は、4、6、7のいずれかになり、演算結果の5は発生しない。

### 排他制御とは

同時に使用できない資源や領域などの1つの対象に対して、複数のアクセス要求があった場合に、アクセスの順序をコントロールして同時にアクセスが起こらないようにすることである。排他制御はマルチタスクシステムでは不可欠な技法の1つである。

### クリティカルセクション(危険領域)

同時に使用できない資源を排他資源、共有される資源を処理する部分をクリティカルセクション(危険領域)という。メモリ上の領域や共有するファイル内のレコードを同時に更新するプログラムの部分はクリティカルセクションである。クリティカルセクションの実行中に、他のクリティカルセクションによる割込の発生を避ける処理が排他制御である。クリティカルセクションの実行中にプリエンプションが発生しないようにすることである。

### 排他制御の方法

ハードウェアで行う方法とソフトウェアで行う方法がある。セマフォはソフトウェアで行う方法の1つである。

セマフォ

セマフォは、セマフォ変数を利用して、P操作、V操作で変数の増減を行い、セマフォ変数の値を参照することによってクリティカルセクションの処理の実行を制御する方式である。タスクがクリティカルセクションに入る前にP操作を実行し、タスクがクリティカルセクションを出るときにV操作を実行する。P操作は、セマフォ変数が0でないなら、変数をデク

リメントして実行を継続し、セマフォ変数が0なら実行が中断されウェイト状態になる。V操作は、セマフォ変数をインクリメントする。セマフォにP操作を実行してウェイトしているタスクがあれば、P操作の再試行を行う。セマフォ変数が0の状態ではクリティカルセクションに入ることができず、待たされる。

#### P操作

セマフォ変数が0でないなら変数をデクリメントして実行を継続する。セマフォ変数が0なら実行が中断されウェイト状態になる。

```
P(S): if (S > 0) then S = S - 1
      クリティカルセクション
      else
        S > 0 となるのを待つ
```

#### V操作

セマフォ変数をインクリメントする。セマフォにP操作を実行してウェイトしているタスクがあれば、P操作の再試行を行う。

```
V(S): if (セマフォ待ちタスクあり) then
      待ちタスクに対してP操作を再試行する
      else
        S = S + 1
```

セマフォによって排他制御するには、バイナリセマフォを使って、タスク間でクリティカルセクションに入る前にP操作を実行し、終了するとV操作を実行する。セマフォは排他資源ごとに作成し、異なるセマフォを使用する。複数のセマフォがある場合、どのセマフォに対する操作かを指示するパラメータ  $n$  を使用する。ジェネラルセマフォはクリティカルセクションに入れるタスク数を複数とる際に使用する。

問題のプロセスの排他制御の仕組みにおける確保状態は、セマフォ変数  $s > 0$  の場合に相当し、解放状態は  $s \leq 0$  に相当する。

同期状態を変更する関数  $l$  は、P操作に該当し、関数  $u$  はV操作に該当する。同期変数  $s$  はセマフォ変数である。

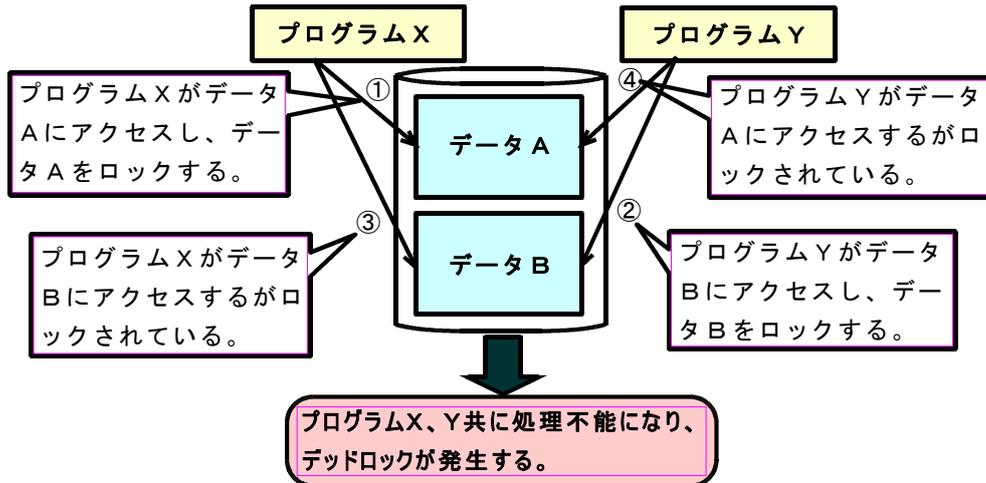
関数  $l$  の操作内容は、 $S$  の状態が解放状態ならば確保状態にし、確保状態ならば解放状態になるまで待ってから確保状態にする。

関数  $u$  の操作内容は、 $S$  の状態が確保状態ならば解放状態にし、解放状態ならば何もしない。

### デッドロック

デッドロックは共有資源を使用する2つ以上のプロセスが、お互いに相手のプロセスが必要としている資源を排他的に使用し、どちらのプロセスもその資源が解放されるのを待ち続けている状態である。

## デッドロックの仕組み



- ① プログラムXがデータAにアクセスし、データAをロックする。
- ② プログラムYがデータBにアクセスし、データBをロックする。
- ③ プログラムXがデータBにアクセスしようとする、プログラムYがロックしているため、アクセスできないで待たされる。
- ④ プログラムYがデータAにアクセスしようとする、プログラムXがロックしているため、アクセスできないで待たされる。
- ⑤ プログラムXもプログラムYも互いに相手の終了を待つ状態になり、デッドロック状態になる。

## 設問1

プロセスP1、P2について、共有データの読み込み、演算実行、結果の書き込みの順序の起こりうるケースについて考えると、解説で説明した4ケースが考えられる。4ケースの演算結果が4、6、7のいずれかになり、演算結果5は実現しない。求める答えはイとなる。

## 設問2

同期変数sはセマフォ変数に相当し、関数lはセマフォのP操作、関数uはセマフォのV操作にそうとする。従って、関数l、関数uの操作内容は解説に示した内容となる。求める答えは、aはア、bはオとなる。

## 設問3

デッドロックが発生するのは、2つのプロセスが、それぞれに相手のプロセスが必要としている共有データをロックしている状態の場合である。プロセスP1が、y1の確保、y2の確保、y2の解放、y1の解放の順に実行する場合、プロセスP2が、y2の確保、y1

の確保、y 1の解放、y 2の解放の順に実行すると、プロセスP 1がy 2の確保、プロセスp 2がy 1の確保の状態で、デッドロックが発生する。求める答えはウとなる。