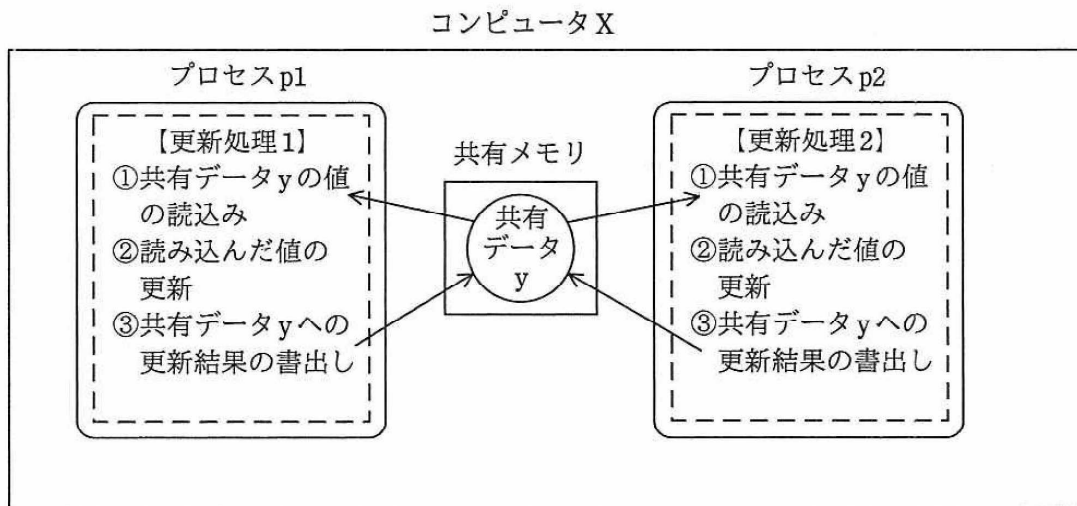


## 問010054問題

プロセスの排他制御に関する次の記述を読んで、設問1～3に答えよ。

単一のCPUと、プロセス間で共有するデータ（以下、共有データという）を格納するためのメモリ（以下、共有メモリという）をもつコンピュータXにおいて、並行に実行される複数のプロセスが、共有データを更新する場合を考える。

二つのプロセスp1、p2が共有データyを更新する際には、図1に示す更新処理1、更新処理2を、それぞれ①～③の順番に行う。



注記 “ $\longrightarrow$ ” は共有データyへのアクセス（読み込み又は書出し）を表す。

図1 二つのプロセスが共有データに対して更新処理を行う例

**設問1** 図1に示すプロセスp1の更新処理1とプロセスp2の更新処理2が、次のとおりに共有データyを更新する場合を考える。

[更新処理1]：共有データyの値を30増加させる。

[更新処理2]：共有データyの値を50減少させる。

二つのプロセスp1、p2の実行前の共有データyの値が100であり、共有データyに対して排他制御を行わずに、プロセスp1が更新処理1を、プロセスp2が更新処理2を並行して各1回だけ行った。結果として、共有データyが取り得ない値を、解答群の中から選べ。

解答群

ア 50

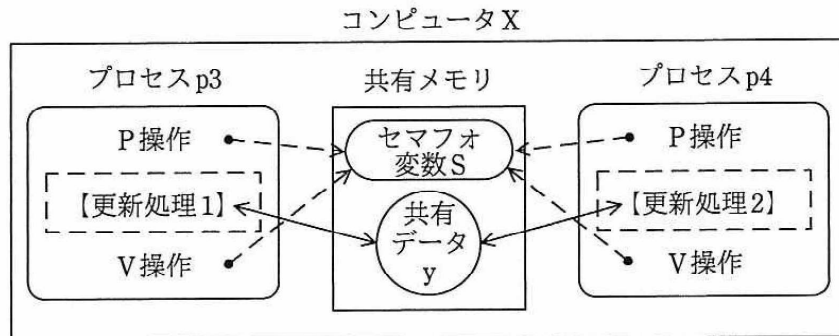
イ 80

ウ 100

エ 130

**設問2** 次の記述中の  に入れる正しい答えを、解答群の中から選べ。

共有データに対して排他制御を行う仕組みとして、セマフォ変数を利用する方法を考える。セマフォ変数は、対応する共有データが、解放されている状態を示す1、又はいずれかのプロセスに確保されている状態を示す0の値をもつ。セマフォ変数の値は、P操作とV操作で変更する。共有データyに対応するセマフォ変数Sを利用したプロセスp3、p4が共有データyを更新する際の動作を、図2に示す。



注記 “ $\longleftrightarrow$ ” は共有データyへのアクセスを表す。  
 “ $\bullet\text{---}\rightarrow$ ” はセマフォ変数Sへの操作を表す。  
 更新処理1と更新処理2は、図1と同じ処理である。

図2 セマフォ変数Sを利用した更新処理

図2において、共有データyに対して排他制御をした上で、更新処理を行うために、プロセスp3は、P操作、更新処理1、V操作の順に処理を行い、プロセスp4は、P操作、更新処理2、V操作の順に処理を行う。P操作は、 a  V操作は、 b 。ここで、P操作とV操作は、それぞれ実行中の中断はないものとする。ただし、プロセスが待ち状態になれば、CPUは別のプロセスを実行させるものとする。

aに関する解答群

- ア セマフォ変数Sの値が0であれば1に変更し、終了する。1であれば0になるまで待った後、1に変更して終了する
- イ セマフォ変数Sの値が0であれば1に変更し、終了する。1であれば何もせずに終了する
- ウ セマフォ変数Sの値が1であれば0に変更し、終了する。0であれば1になるまで待った後、0に変更して終了する
- エ セマフォ変数Sの値が1であれば0に変更し、終了する。0であれば何もせずに終了する

bに関する解答群

- ア セマフォ変数Sの値が0であれば1になるまで待った後、終了する
- イ セマフォ変数Sの値が1であれば0になるまで待った後、終了する
- ウ セマフォ変数Sの値を0にして終了する
- エ セマフォ変数Sの値を1にして終了する

**設問3** プロセス p 5, p 6が更新する二つの共有データ y 1, y 2があり, それぞれに異なるセマフォ変数を対応させて, 個別に確保と解放ができるようにした場合を考える。共有データ y 1に対してセマフォ変数 S 1を, 共有データ y 2に対してセマフォ変数 S 2を用いて排他制御を行う。プロセス p 5が, 次の順序で共有データ y 1, y 2に対する確保と解放を行うとき, プロセス p 6が, 共有データ y 1, y 2に対する確保と解放を行う順序によってはデッドロックが発生する可能性がある。デッドロックが発生する可能性がある, プロセス p 6の共有データ y 1, y 2に対する確保と解放の順序を, 解答群の中から選べ。

[プロセス p 5の共有データに対する確保と解放の順序]

y 1の確保, y 2の確保, y 2の解放, y 1の解放

解答群

- ア y 1の確保, y 1の解放, y 2の確保, y 2の解放
- イ y 1の確保, y 2の確保, y 1の解放, y 2の解放
- ウ y 1の確保, y 2の確保, y 2の解放, y 1の解放
- エ y 2の確保, y 1の確保, y 1の解放, y 2の解放
- オ y 2の確保, y 2の解放, y 1の確保, y 1の解放