

④並行プロセスと状態遷移

高度OS2013年度

問1 プロセスの状態遷移

新規状態のプロセスで生成が完了し、レディ状態となった。その後、以下の順番でこのプロセスに事象が発生した。OSによる状態遷移処理後のプロセスの状態は何か。

(1)CPU割当て、(2)プリエンプシジョン、(3) CPU割当て、(4)入出力要求、(5)入出力完了

【(1)～(5)の順で各角のコンマ「,」で区切り、解答欄に全角文字で記入。角、空白などの余計な文字、半角コンマ「,」や読点「.」などを用いると不正解となる。〈問は出題も問解もなので注意すること〉

実行中、レディ、実行中、待機、レディ

発生した事象	状態
(新規状態で生成が完了)	レディ
(1)CPU割当て	実行中
(2)プリエンプシジョン	レディ
(3)CPU割当て	実行中
(4)入出力要求	待機
(5)入出力完了	レディ



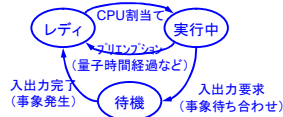
問2 プロセスの状態遷移

以下に示す状態遷移で、有り得ないものはどれか(複数解答)。

- A. 実行中状態のプロセスが入出力命令を発行した。
- B. CPUを割当てられた待機状態のプロセスが命令の実行を開始した。**
- C. 実行中のプロセスからCPUを取り上げた。
- D. レディのプロセスが発行していた入出力命令が終わった。**
- E. 入出力が完了してレディ状態になった。

- A: 実行中に入出力命令の発行＝入出力要求(OK)
(入出力要求が発生し得るのは実行中状態のみ)
- B: 待機状態のプロセスにはCPUを割当てない
(事象が発生するまで実行できない)(NG)
- C: 実行中状態でプリエンプシジョン(OK)
- D: 入出力命令の発行＝入出力要求
レディ状態ではなく待機状態のはず(NG)
- E: 入出力要求を行って待機状態のプロセスに
おいて入出力が完了し、レディ状態となる(OK)

各状態において発生し得る事象を理解すること



問3 並行プロセスの状態遷移

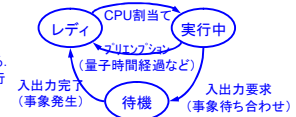
システム内に2つのプロセスP1, P2が並行して動作している(他にはプロセスは無い)。P1:レディ, P2:実行中のときに、P2が入出力要求を行った。OSの状態遷移処理が終わったあとのプロセスの状態は何か。

答 実行中、待機

P2は実行中に入出力要求を行った。
(入出力のためのシステムコール命令を実行した)
そのため、OSはP2を待機状態にする。
(入出力が終わるまで、次の命令を実行できない。)

その結果CPUが空いたので、OSはレディ状態のP1
にCPU割り当てを行う。そのため、P1は実行中となる。
(レディ状態のプロセスはCPUが割当てられれば実行
できる。)

OSはCPUを有効利用するために、CPUが空けば、レ
ディ状態のプロセスにCPUを割当てて、実行中状態
にする。



問4 並行プロセスの状態遷移

システム内に2つのプロセスP1, P2が並行して動作している(他にはプロセスは無い)。P1:待機, P2:実行中のときに、P2が入出力要求を行った。OSの状態遷移処理が終わったあとのプロセスの状態は何か。

答 待機、待機

P2は実行中に入出力要求を行った。
(入出力のためのシステムコール命令を実行した)
そのため、OSはP2を待機状態にする。

その結果CPUが空くが、レディ状態のプロセスが無い
ので、CPUを割当てない。

例えば、入出力のためのシステムコール命令を実行
したプロセスは、入出力が終わるまで、次の命令を実
行できない(このプロセスにCPUを割当てても無駄)。
待機状態は、プロセスが実行できないことを表すため
の状態である。
(待機状態のプロセスにはCPUを割当てない)



問5 並行プロセスの状態遷移

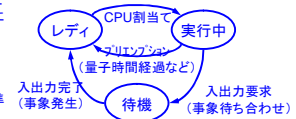
システム内に2つのプロセスP1, P2が並行して動作している(他にはプロセスは無い)。P1:待機, P2:待機のときに、P2の入出力処理が終わった。OSの状態遷移処理が終わったあとのプロセスの状態は何か。

答 待機、実行中

P2が待機状態のときに入出力完了の事象が発生した。
そのため、OSはP2をレディ状態にする。

また、CPUが空いているので、OSは今、レディ状態と
なったP2にCPUを割当てて、その結果P2が実行中にな
る。

入出力のためのシステムコール命令を実行したプロ
セスは、入出力が終わると、次の命令が実行可能に
なる。レディ状態は、プロセスがこうに実行する準
備ができていないことを表すための状態である。
(レディ＝ready:用意が整って、準備ができて)



スライド(添付ファイル:問6～10)

セマフォの定義
P(S):
Sの値を1減らす;
S ≥ 0 → nop;
S < 0 → 発行元プロセスを待機中に;
戻る;

V(S):
Sの値を1増やす;
S > 0 → nop;
S ≤ 0 → 待機中プロセスをレディに;
戻る;

消費者:(以下を繰り返す)
(1)P(S2);
(2)nextc = buffer[out];
(3)out=(out+1) % n;
(4)V(S1);
(5)nextcのデータをプリンタにwrite;

生産者:(以下を繰り返す)
(6)ディスクreadし, nextipに入力;
(7)P(S1);
(8)buffer[in]=nextip;
(9)in=(in+1) % n;
(10)V(S2);

問6 セマフォ変数の初期値

スライド【添付ファイル】の消費者プロセス、生産者プロセスの排他制御をセマフォ変数S1, S2によって行う。S1, S2の初期値は幾つか、但し、以降の問題を含めて、buffer[]の数は2(n=2)で、開始時は空きとする。【S1, S2の値は整数値を全角のコンマ「,」で区切り、半角数字で横書きに記入】

答 2, 0

セマフォ変数の初期値は使用可能な共有資源数である。
この問題の共有資源はバッファ(buffer[])である。

S2は消費者が使用する資源で入力済みバッファ数(fullに相当)、
S1は生産者が使用する資源で空きバッファ数(emptyに相当)
開始時点では、バッファは2つとも空きなので、空きバッファが2
(S1=2, 入力済みバッファが0(S2=0)。

問7 セマフォによる状態遷移

スライド【添付ファイル】のプログラムにおいて、生産者、消費者の生成が完了し、消費者にCPUが割り当てられ、消費者は(1)を実行した。OSの状態遷移処理が終わった時点でのプロセスの状態は何になるか。【生産者、消費者の値は整数値を全角のコンマ「,」で区切り、横書きに記入。】

答 実行中, 待機

	生産者	消費者	S1	S2
生産者、消費者生成	レディ	レディ	2	0
消費者にCPU割当て	レディ	実行中	2	0
(1)を実行		待機	2	-1
、CPU割当て	実行中			

消費者による(1)P(S2)の実行により、S2=-1となる。OSは消費者を待機状態にする。(消費者が使う入力済みバッファが無いので、消費者の実行を止めなければならない)その結果、CPUが空くので、OSはレディ状態の生産者を実行中にする。即ち、OSの状態遷移処理が終わった時点での生産者の状態は実行中である。(ただし、次の(6)の実行によってすぐに待機状態になる)

問8 セマフォによる状態遷移

前問に続いて、生産者が(6)を実行した。続いて生産者のreadが完了して生産者が再び実行中になり、(7)～(10)を実行した。(10)に対するOSの処理が終わった後、生産者、消費者の状態とセマフォ変数の値はどのようになっているか。【生産者、消費者、S1, S2の値は整数値を全角のコンマ「,」で区切り、横書きに記入。状態名は漢字または全角カタカナ、変数値は半角数字。】

答 実行中, レディ, 1, 0

	生産者	消費者	S1	S2
生産者、消費者生成	レディ	レディ	2	0
消費者にCPU割り当て	レディ	実行中	2	0
(1)を実行、CPU割当て	実行中	待機	2	-1
(6)を実行(read発行)	待機	待機	2	-1
read完了。(7)～(10)を実行	実行中	レディ	1	0

生産者は(6)の実行により待機状態となる。
生産者は、read完了により、一度レディ状態となるが、CPUが空いているので、実行中となる。
生産者による(10)のV(S2)の実行により、S2=0となったため、OSは消費者をレディ状態にする。
生産者はまだ実行中のままである(この後、(6)を実行すると待機状態になる)。

問9 セマフォによる排他制御

前問に続いて、生産者が(6)を実行し、続いて消費者が(2)～(4)を実行し、(5)システムコールを発行した。(5)のシステムコール発行に対するOSの処理が終わった後、生産者、消費者の状態とセマフォ変数の値はどのようになるか。【生産者、消費者、S1, S2の値は整数値を全角のコンマ「,」で区切り、横書きに記入。状態名は漢字または全角カタカナ、変数値は半角数字。】

答 待機, 待機, 2, 0

	生産者	消費者	S1	S2
生産者、消費者生成	レディ	レディ	2	0
消費者にCPU割当て	レディ	実行中	2	0
(1)を実行、CPU割当て	実行中	待機	2	-1
(6)を実行(read発行)	待機	待機	2	-1
read完了。(7)～(10)を実行	実行中	レディ	1	0
(6)を実行、CPU割当て	待機	実行中	1	0
(2)～(4)を実行	待機	実行中	2	0
(5)を実行	待機	待機	2	0

生産者は(6)の実行(システムコールを発行)により待機状態となる(CPUが空く)。レディ状態の消費者は、CPUが割当てられて実行中となり、最後に実行した(1)に続く(2)～(4)を実行する。さらに、(5)の実行(システムコール発行)により待機状態となる。

問10の状態になるまでの処理

問9に続いて、(A)生産者、(B)生産者の順で実行中になると、問10の始めの状態になる。問10では、この後、(C)生産者が実行中になると、状態遷移がどのようになるかを質問している。

	生産者	消費者	S1	S2
問9の処理が終わった時点	待機	待機	2	0
(A)read完了。	実行中	待機	2	0
(A)続いて(7)～(10)を実行	実行中	待機	1	1
(A)続いて(6)を実行	待機	待機	1	1
(B)read完了。	実行中	待機	1	1
(B)続いて(7)～(10)を実行	実行中	待機	0	2
(B)続いて(6)を実行 【問10の開始時点】	待機	待機	0	2
(C)read完了 【生産者が実行中になる】	実行中	待機	0	2
(C)続いて(7)を実行				

これで生産者の実行中状態が終わる

問10 セマフォによる排他制御

消費者、生産者は、(1), (10)の実行によって待機状態にあり、S1=0, S2=2とする
(問9の直後ではない)。この後、生産者が実行中状態となった。生産者の実行が
終わった後(OSの状態遷移処理が終わった後)で、生産者、消費者の状態とセマ
フォ変数の値はどのようなになるか。【生産者、消費者、S1、S2の欄に状態名と変数値を全角のコンマ「,」で区切り、解答欄に記入。状態名は
漢字または全角カタカナ。変数値は半角数字。】

答 待機, 待機, -1, 2

	生産者	消費者	S1	S2
(B)続いて(6)を実行【問10開始時点】	待機	待機	0	2
(C)read完了。	実行中	待機	0	2
(C)続いて(7)を実行	待機	待機	-1	2
問10 以後の処理				
(D)write完了。	待機	実行中	-1	2
(D)続いて(2)~(3)を実行	待機	実行中	-1	1
(D)続いて(4)を実行	レディ	実行中	0	1
(D)続いて(5)を実行	実行中	待機	0	1
(E)続いて(8)~(10)を実行	実行中	待機	0	2