

基礎OS⑧ プロセス管理(1)

2012年度(3時限目)

問1, 2

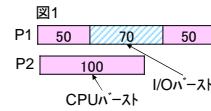
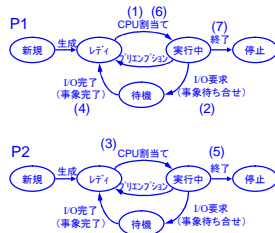
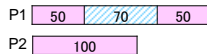


表1

時刻	発生事象	P1	P2
—	(P1,P2生成)		
0	(1)CPU割当て(P1)		
	(2)I/O要求 (3)CPU割当て		
	(4)I/O完了		
	(5)終了		
	(6)CPU割当て		
	(7)終了		

問1 事象の発生と状態遷移

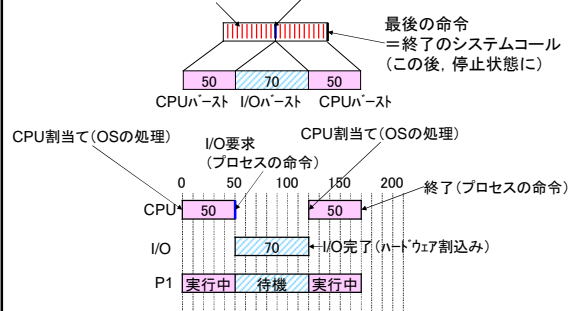
スライド【資料ファイル】図1のプロセスを並行して実行させたと、時刻0のP1における事象(1)に続き、表1のような事象が発生した。事象(2)~(7)は、どのプロセスにおける事象か。



時刻	発生事象	P1	P2
—	(P1,P2生成)	レディ	レディ
0	(1)CPU割当て(P1)	実行中	レディ
	(2)I/O要求	待機	実行中
	(3)CPU割当て	待機	実行中
	(4)I/O完了	レディ	実行中
	(5)終了	レディ	停止
	(6)CPU割当て	実行中	停止
	(7)終了	停止	停止

命令の実行と状態遷移

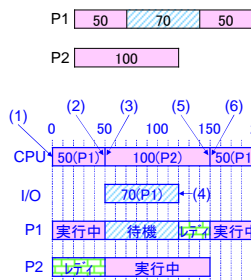
通常の命令 (CPUを使って計算) I/O要求の命令=システムコール (I/Oバースト開始. I/O完了までCPUを使わない)



問2 事象の発生と状態遷移

問1のプロセスに対する事象(2), (4), (5), (7)の発生時刻は?

答 50,120,150,200



時刻	発生事象	P1	P2
—	(P1,P2生成)	レディ	レディ
0	(1)CPU割当て(P1)	実行中	レディ
50	(2)I/O要求	待機	実行中
	(3)CPU割当て	待機	実行中
120	(4)I/O完了	レディ	実行中
150	(5)終了	レディ	停止
	(6)CPU割当て	実行中	停止
200	(7)終了	停止	停止

時刻は作図しながら求める

問3 プロセスの状態遷移

プロセスの生成から終了までの状態遷移として、あり得ないものはどれか。(第2種 平成10年度・春期)

- A. (生成)→レディ状態→実行中状態→(終了)
- ⓑ. (生成)→レディ状態→待機状態→実行中状態→(終了)
- C. (生成)→レディ状態→実行中状態→レディ状態→実行中状態→(終了)
- D. (生成)→レディ状態→実行中状態→待機状態→レディ状態→実行中状態→(終了)

レディ: CPU待ち
実行中: CPU使用中
待機: 入出力中
基本: レディ→実行中→待機→レディ. プリエンジョン: 実行中→レディ
レディから待機にはならない. 待機から実行中にはならない

問4 状態遷移の事象

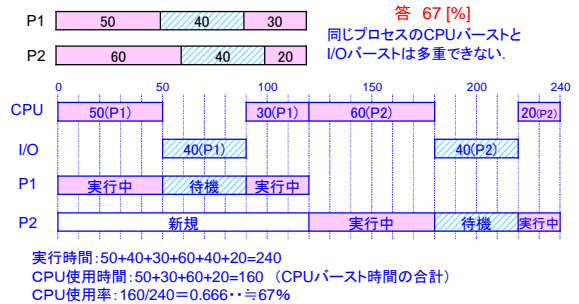
プロセスの状態遷移において、実行中状態から待機状態に遷移する事象は、以下のどれか？

- A. 不正命令の実行
- ☒ B. ファイル入力要求の実行
- C. 関数呼び出しの実行
- D. 磁気テープの読み込み終了
- E. 磁気テープ装置の故障
- F. 時計割込み

I/O要求のシステムコールを発行すると待機状態に遷移する。
その入出力が終了すると、レディ状態に遷移する。
例えば、プロセスが、ファイル出力要求のシステムコールを発行すると待機状態に遷移し、OSは磁気ディスクに対し、書き込み指示を行う。
書き込みが終了すると出力完了の割り込みが発生し、レディ状態となる。

問5 プロセスの実行(1)

単独で実行するとスライド図2.11の動作をするプロセスP1、P2がある。これらを多重せずに、順次実行させる。開始から終了までのCPU使用率[%]を求めよ。【数値(整数値)のみ半角数字で記入。%記号の付加点以下四捨五入。】



問5における事象の発生と状態

時刻	発生した事象	P1	P2
	(P1生成)	レディ	新規
0	CPU割当て(P1)	実行中	新規
50	I/O要求	待機	新規
90	I/O完了	レディ	新規
	CPU割当て(注1)	実行中	新規
120	終了	停止	新規
	P2生成(注2)		レディ
	CPU割り当て(注1)		実行中
180	I/O要求		待機
220	I/O完了		レディ
	CPU割当て(注1)		実行中
240	終了		停止

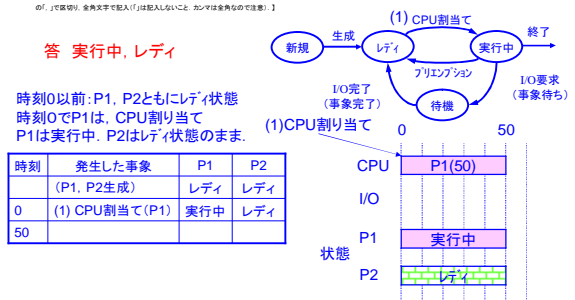
I/O完了や生成により、瞬間的に、レディ状態となるが、直後のCPU割当てにより、実行中状態になる。

注1: 実行中のプロセスが無い時に、プロセスがレディ状態となるような事象が発生した場合、OSは「CPU割当て」を行い、そのプロセスを実行中状態にする(プロセスは瞬間的にレディ状態となるが、即座に実行中状態になる)。
注2: P1が終了したので次のプロセスを生成している(順次実行であるため)

問6 プロセスの実行(2)

問5のプロセスを多重して動作させる。時刻0以前にプロセスP1、P2ともに生成が終了しており、実行待ち列に並んでいる。時刻0で、P1にCPUが割り当てられた。時刻0以後のP1、P2の状態は何か。【P1はP1、P2の欄に状態名のみを各角の「」で書き、全角文字で記入(「」は記入しないこと。カンマは全角のもので注意)】

答 実行中, レディ

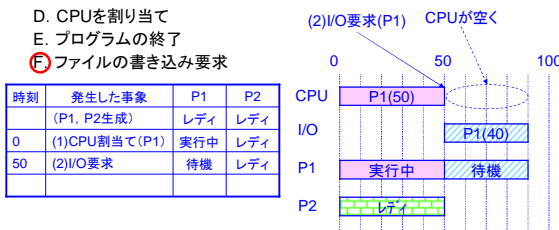


問7 プロセスの実行(3)

時刻50からプロセスP1は、I/Oバーストを始める。その直前に、プロセスP1のプログラムが実行した処理として考えられるものは以下のどれか。

- A. プロセスの生成
- B. 待機事象が発生
- C. プリエンブション
- D. CPUを割り当て
- E. プログラムの終了
- ☒ F. ファイルの書き込み要求

I/O要求のシステムコールによりI/Oバーストが開始される。(即ち、CPUバーストの最後の命令がI/O要求) 選択肢の中でI/O要求に相当するのは「F」。(P1は待機状態になり、実行中のプロセスが無くなる)

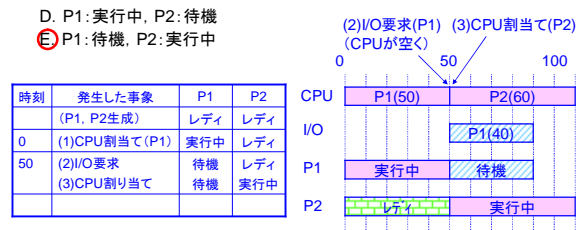


問7(解説問題)

時刻50に、プロセスP1はI/O要求のシステムコールを発行し、I/Oバーストを始める。その結果、P1、P2の状態は以下のどれになるか。

- A. P1: 新規, P2: 停止
- B. P1: レディ, P2: 実行中
- C. P1: 実行中, P2: レディ
- D. P1: 実行中, P2: 待機
- ☒ E. P1: 待機, P2: 実行中

CPUが空いたので、OSはレディ状態のP2にCPU割当てを行う。

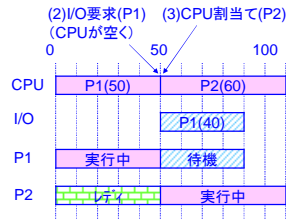


問8 プロセスの実行(4)

時刻50においてP1がI/Oバーストを始めると、CPUが空く。OSはCPUをP2に割当てて(CPUが切り替わり、P1は待機、P2は実行中になる)。このような動作を示す用語は、以下のどれか。

- A. CPU切り替え
- B. CPUスケジューリング
- C. 実行制御
- D. レジスタ退避
- E. プロセス切り替え
- F. コンテキスト切り替え**

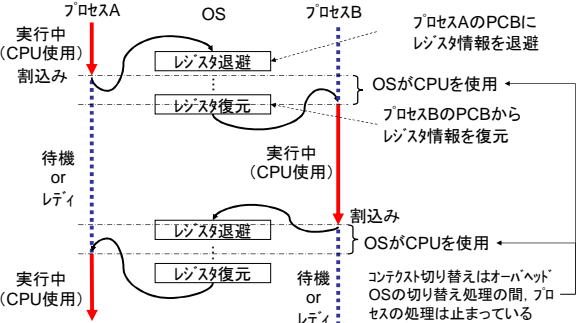
CPUが空いたので、OSはP2にCPUを割当てて。このようなCPUの切替をコンテキスト切替と言う。



コンテキスト切替(コンテキストスイッチとも言う)

実行中状態のプロセスから他のプロセスにCPUを切り替える

context: 状況、文脈→実行状況の切り替え、処理の流れ(文脈)の切り替え



問9 プロセスの実行(5)

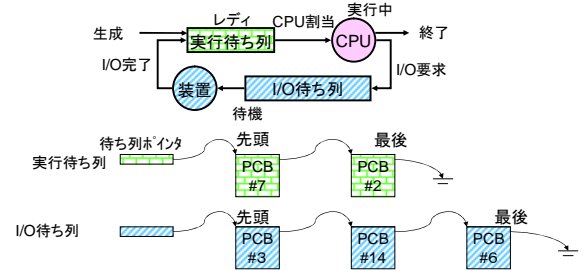
P1は時刻50から時間40だけI/Oバーストを行い、時刻90にI/O完了する。しかし、P2がまだCPUを使用している(実行中状態)。このときのP1の動きとして適当なものは以下のどれか。

- A. I/O待ち列で待つ
- B. 新たなプロセスを生成する
- C. 待機状態になる
- D. 実行を終了する
- E. 実行待ち列で待つ**



実行待ち列とI/O待ち列

実行待ち列: レディ状態のプロセスが、CPUが割り当てられるまでの待ち列
I/O待ち列: I/O要求をした待機状態のプロセスが装置が空くまでの待ち列(注)
 待ち列は、PCBのリストにより表される。

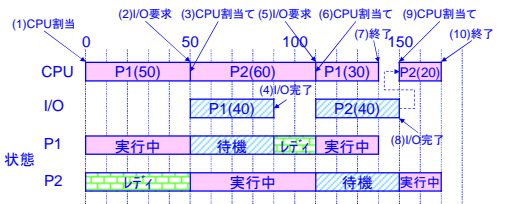
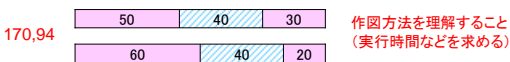


注: 実際にはI/O待ち列は、I/O装置毎に作られる。以下、簡略化のため1装置の場合で説明

問10 プロセスの実行(2)

問5の2プロセスを多重して実行させる。2プロセスの実行時間(開始から終了までの時間)とその間のCPU使用率[%]を求めよ。【実行時間: CPU使用率(%)の小数点以下は四捨五入の後に半角以下(「0」で切り、半角数字で記入(「1」は記入しないこと。)]

答 170,94



CPU使用時間=50+60+30+20=160(問5と同じ)
 CPU使用率=160/170=0.941... ≈94[%]

問10における事象の発生と状態

時刻	発生した事象	P1	P2
	(P1,P2生成)	レディ	レディ
0	(1)CPU割当て(P1)	実行中	レディ
50	(2)I/O要求	待機	実行中
	(3)CPU割当て		
90	(4)I/O完了	レディ	実行中
110	(5)I/O要求	待機	実行中
	(6)CPU割り当て	実行中	実行中
140	(7)終了	停止	
150	(8)I/O完了		実行中
	(9)CPU割当て(注2)		実行中
170	(10)終了		停止

注1: 実行中のプロセスの状態が待機または停止状態になると、OSはレディ状態のプロセスにCPUを割当てる(レディ状態のプロセスを実行中状態にする)。
 注2: 実行中のプロセスが無い時に、プロセスがレディ状態になると、OSはそのプロセスにCPU割当てを行う(プロセスは瞬間的にレディ状態となるが、即座に実行中状態になる)。