

問3 OSにおけるプロセスのスケジューリング(ソフトウェア) (H26 秋・FE 午後問3)

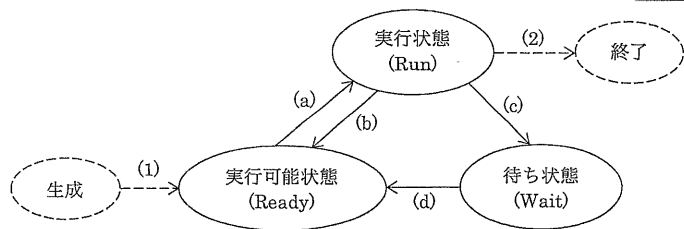
【解答】
[設問1] ウ
[設問2] イ
[設問3] aーキ, bーカ, cーウ

【解説】
OSの機能の一つにプロセスのスケジューリングがある。なお、プロセスの代わりにタスクと記述されているテキストもあるが、同じ意味である。
これまでの午前・午後問題では、単一のスケジューリング方式(優先度方式、到着順方式、ラウンドロビン方式)について、CPUの処理時間と入出力の処理時間を割り当てて、全体の処理時間(ターンアラウンドタイム)を計算させる問題が多かった。本問は、幾つかのスケジューリング方式を一括して取り上げ、解答させるという総合的な問題となっている。

具体的には、設問ごとにスケジューリング方式が分かれており、次のようになっている。

- (1) 到着順方式 [設問1]
プロセスを到着した順に待ち行列の末尾に登録し、待ち行列の先頭のプロセスから順にCPUを割り当てて実行する方式
- (2) ラウンドロビン方式 [設問2]
プロセスを到着した順に待ち行列の末尾に登録し、実行中のプロセスがある一定時間(タイムクォンタム)経過した後に、そのプロセスの実行を中断して待ち行列の末尾に移動し、待ち行列の先頭のプロセスから順にCPUを割り当てて実行する方式
- (3) 残余処理時間順方式 [設問3]
一定時間ごとに、実行状態と実行可能状態にあるプロセスの残余処理時間(プロセスの残りの処理が完了するまでに必要なCPUの割当て時間)を比較し、その時間が最も短いプロセスから順にCPUを割り当てて実行する方式

ここで問題文にも記述されているが、プロセスの状態の遷移の流れ(図3)を確認する。



図A プロセスの状態の遷移

生成されたプロセスは、(1)の矢印に沿って実行可能状態に遷移する。つまり、全てのプロセスは生成されると同時に実行可能状態になるのである。

次に、実行可能状態にある幾つかのプロセスの中から一つを取り出し、(a)に沿って実行状態にする。ここで実行可能状態にあるどのプロセスを実行状態にするかは、スケジューリング方式によって決まる。

入出力の発生によってCPUの実行が中断されたときは、(c)に沿ってプロセスは待ち状態になる。入出力が完了したら、(d)に沿ってプロセスは実行可能状態に戻る。

実行が終了したプロセスは、(2)の破線に沿って終了状態になる。本問では(1)や(2)の遷移は問われていない。

なお、(b)の遷移は実行状態のプロセスを強制的に実行可能状態に引き戻す操作を指しており、これをプリエンプションという。これは、(2)のラウンドロビン方式や(3)の残余処理時間順方式、優先度方式(本問では出てこない)に必要な仕組みである。

[設問1]
到着順方式によるプロセスの状態の遷移についての設問である。
プロセスの状態の遷移は問題文にも説明があるが、基本を理解しているとイメージがつかみやすい。

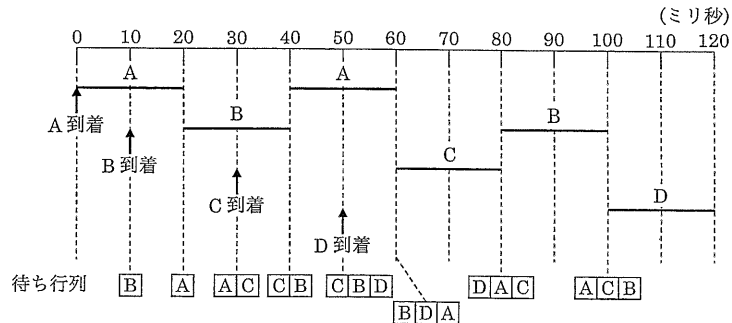
プロセスXは図4の順序で処理が行われる。ここで、「CPUを使用」と記述された部分では、プロセスXは実行状態になっている。データの入力やデータの出力の部分では入出力の処理であるから、プロセスXは待ち状態になる。次に、プロセスXの状態と状態の遷移を表にして示す。

表A プロセスXの状態の遷移		
プロセスXの処理	状態	状態の遷移
①前処理(CPUを使用)	実行可能	(a)
	実行	
②データの入力 開始	待ち	(c)
	実行可能	(d)
③計算処理(CPUを使用)	実行	(a)
	待ち	(c)
④データの出力 開始	待ち	(c)
	実行可能	(d)
⑤後処理(CPUを使用)	実行	(a)

この表に示しているとおり、状態の遷移は(a)→(c)→(d)→(a)→(c)→(d)→(a)の順に起こる。したがって、
(a)……3回
(b)……0回
(c)……2回
(d)……2回
となり、(ウ)が正解である。

[設問2]
ラウンドロビン方式によるプロセスの待ち行列への登録順序を答える設問である。
各プロセスの到着時刻とタイムクォンタムを基に待ち行列の中身を変えていかな

ければならず、タイムチャートをうまく描くことが解答を導くためのポイントになる。
次のように、10ミリ秒ごとの目盛で図を描いてみよう。



図B プロセスA~Dの実行過程

なお、実行状態にあるプロセスを実線の上に示している。また、待ち行列は左側が先頭を示す。この結果から、時刻100ミリ秒の時点でプロセスDが実行を開始し、このときの待ち行列は先頭からA, C, Bの順になっていることが分かる。
したがって、(イ)が正解である。

[設問3]
残余処理時間順方式によるプロセスのターンアラウンドタイムの計算の問題である。問題文にあるとおり、残余処理時間とは、その時点でプロセスの残りの処理が完了するまでに必要なCPUの処理時間のことである。
a, b, c: プロセスAが登録された時点(時刻0)として10ミリ秒ごとに表1の各プロセスの残余処理時間を比較し、最も時間の短いプロセスを実行状態に移せばよい。このとき、実行状態にあるプロセスも比較の対象とするため、もし実行中のプロセスよりも残余処理時間が短いプロセスが実行可能状態にある場合は、実行中のプロセスを実行可能状態に強制的に移行させなければならない。つまり、プリエンプション(図Aの(b)の遷移)が起こる。
このことと表1の到着時刻、処理時間を考慮して図5の先を続けて記入していくことにする。

	30	60	90	120	150	180	210	240	270	300	
A	○	○	○	○	○	○	○	○	○	○	○
B	○	○	○	○	○	○	○	○	○	○	○
C	○	○	○	○	○	○	○	○	○	○	○
D											

図C 問題文図5を埋めたもの

なお、図中の縦棒は、プロセスの開始と終了を表し、プロセスの終了時刻はendの下に記述してある。

プロセスの切替え時に残余処理時間の比較をしなければならないが、これに掛かる時間(オーバヘッド)は考えないものとするという記述がある。したがって、プロセスは瞬間的に切り替わると考える。

いま、時刻10ミリ秒(プロセスAが開始してから経過時間)の時点で考えると、プロセスBが到着して実行可能状態になり、実行状態にあるプロセスAとの間で残余処理時間を比較する。プロセスAは10ミリ秒だけ実行したので、残余処理時間は120-10=110ミリ秒、プロセスBの残余処理時間は90ミリ秒である。したがって、プロセスBが実行状態になり、プロセスAは実行可能状態に戻される。

同様に、時刻30ミリ秒の時点でプロセスA(実行可能)、B(実行)、C(実行可能)の三つを比較すると、プロセスAの残余処理時間は110ミリ秒、プロセスBの残余処理時間は90-20=70ミリ秒、プロセスCの残余処理時間は60ミリ秒であるから、プロセスCが実行状態になり、プロセスA, Bが実行可能状態になる。

同様に、時刻50ミリ秒の時点ではプロセスDが新たに実行可能状態になり、プロセスA, B, C, Dを全て比較する。プロセスAの残余処理時間は110ミリ秒、プロセスBの残余処理時間は70ミリ秒、プロセスCの残余処理時間は60-20=40ミリ秒、プロセスDの残余処理時間は30ミリ秒であるから、プロセスDが実行状態に遷移し、プロセスA, B, Cが実行可能状態になる。プロセスDの実行開始から10ミリ秒が経った60ミリ秒目の時点で、再度、残余処理時間を比較する。その結果、プロセスA, B, Cの残余処理時間は前述した50ミリ秒目時点と変わらず、プロセスDの残余処理時間だけ10ミリ秒減って、20ミリ秒になる。その結果、プロセスDの処理が続く。これは70ミリ秒目でも同様である。

プロセスDは時刻80ミリ秒で終了する。この後は残余処理時間が少ない順に実行状態に遷移し、実行終了まで実行を続けると考えてよく、プロセスC(40ミリ秒)、プロセスB(70ミリ秒)、プロセスA(110ミリ秒)の順で実行される。()内は時刻80ミリ秒の時点での各プロセスの残余処理時間である。

したがって、図Cのようになり、プロセスCの終了時刻は80+40=120ミリ秒、プロセスBの終了時刻は120+70=190ミリ秒、プロセスAの終了時刻は190+110=300ミリ秒となる。よって、各プロセスのターンアラウンドタイムは次のようになる。

プロセスA : 300ミリ秒
プロセスB : 190-10=180ミリ秒
プロセスC : 120-30=90ミリ秒
したがって、空欄aは(キ)、空欄bは(カ)、空欄cは(ウ)が正解である。