

【雜考】

設問1] a-オ, b-ウ

【設問2】 c-ウ, d-オ

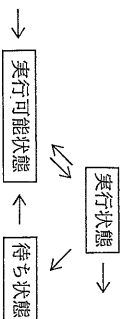
【解説】

CPUの割当て方式に関する問題である。この題材は、午前問題の出題範囲に含まれているので、各方式について理解しているはずである。しかし、たとえば理解に自信がなくても、説明に沿って考えていけば解ける問題である。この際、確実に理解しておくとよいだろう。

その方式として、まず、到着順方式とラウンジ方式の説明がされている。

到着順方式は、文字通り到着順に実行していく方式である。到着順にプロセスを実行の待ち行列の末尾に登録し、実行中のプロセスが終了したら、待ち行列の先頭からプロセスを一つ取り出して実行を開始するという手順である。一つのプロセスの実行が始まったら、そのプロセスが終了するまで CPU を割り当てられるので、処理時間の長いプロセスがあると、後続のプロセスは待たされることになる。

ところで、問題の説明中に、「待ち行列に登録されているプロセスの状態を実行可能状態、実行中のプロセスの状態を実行状態と呼ぶ」とある。「実行可能状態」、「実行状態」に加え、プロセスの出力要求を出して、その入出力処理の終了待ちの状態のことである。プロセスが入出力要求を出した場合、この「待ち状態」は考慮しなくてよい。「実行可能状態」と「実行状態」の行き来だけを考えればよい。



ラウンドロビン方式は、幾つものプロセスが、一定時間（タイムクォンタム）ずつ、いわば輪番制で CPU を割り当てられる方式である。到着順にプロセスを実行の待ち行列の末尾に登録するのは、到着順方式と同じである。プロセスの実行は、待ち行列の先頭から一つ取り出し開始するが、タイムクォンタムだけ経過すると、実執行を中断し、そのプロセスは待ち行列の末尾に再登録する。次は、待ち行列の先頭からプロセスを一つ取り出し実行を開始するのである。タイムクォンタム以内で終了すれば、もう待ち行列に再登録する必要はない。この方式では、遅れてきたプロセスでも、必ず実行の順番が回ってくる。

このような実行順序の決定方式の効果を示す指標として、ターンアラウンドタイムがある。ターンアラウンドタイムは、プロセスが待ち行列に到着してから実行が終了するまでの時間である。

[設問1]

ここでは、表1に示される四つのプロセスについて、両方式で実行した場合のタイムパフォーマンスの平均値が問われている。平均値ということで、方式ごとに四つすべてのターンアラウンドタイムを調べることになる。頭の中だけで考えて、解答かき写しをするならばそれにしたことはない。しかし、このような問題では、自分自身からオペレーティングシステムにしたことにより、各方式の手順を守りつつ、タイムテーブルのような図を描いて考えとよい。プロセスの状態（実行・実行可能状態）、待ち行列列の並びなどが、はつきりしてきてくるだろう。試験会場では方眼用紙も定規もないが、大ざっぱであればいいので、線を引いて、実行過程、実行時間など必要と思われる情報をメモ書きしていくとよい。

なお、設問中に「プロセスの取出し、及び中断の処理でのオーバヘッドは考えない」とあるのは、実行に要する時間だけ考えればよいことである。実際には、待ち行列に関して取出しや再登録、プロセスごとの情報退避や再設定などに時間を要するはずであるが、問題を解く上では、それらを考慮しなくてよいことである。また、「CPUを割り当てられたプロセスは、タイムクォンタム以外で中断することはない」とあるのは、例えば、データの入出力等で実行を中断することはないということである。これらの注意書きは、問題を簡略化して考えればよいという配慮である。

空欄 a : 到着順方式の場合のターニングラウンドタイム平均値を求める。

四つのプロセスの待ち行列への到着順は、A、A の 10 ミリ秒後に B、A の 30 ミリ秒後に C、A の 50 ミリ秒後に D、の順である。待ち行列に登録された時刻から、ターニングラウンドタイムを計算することになる。タイムテーブルを描くと、次のようになる。これは、あくまでタイムテーブルの例であって自分なりの描き方だよ。

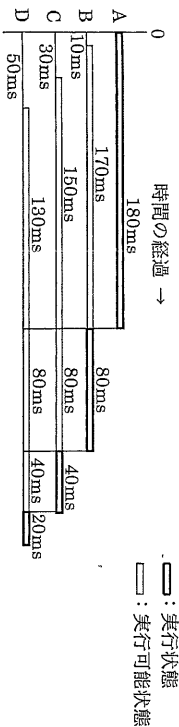


図 1 到着順方式のタイムテーブル例

[タイムテーブルの解説]

A:「Aが到着したとき、実行可能状態及び実行状態のプロセスはない」ということから、Aが待ち行列に到着するとすぐに実行できることになる。したがって、Aのターンアラウンドタイムは、実行時間の180ミリ秒だけである。B: Aが実行を開始して10ミリ秒後に到着する。そのときは、まだAの実行中である。Bが実行できるのは、Aの実行が終了してからである。つまり、IはAの実行(180ミリ秒)が終わるまで、待ち行列で170(180-10)ミリ秒実行を待ち、その後実行(80ミリ秒)を開始する。したがって、Bのターンアラウンドタイムは、 $170+80=250$ (ミリ秒)である。

C: Aが実行を開始して30ミリ秒後に到着する。そのときは、まだAは実行中である。Cが実行できるのは、Aの次にBが実行を終了した後である。つまり、Aの実行終了までの150(180-30)ミリ秒、Bの実行終了まで80ミリ秒待つ、実行(40ミリ秒)を開始する。したがって、Cのターンアラウンドタイムは、150+80+40=270(ミリ秒)である。

D: Aが実行を開始して50ミリ秒後に到着する。そのときは、まだAは実行中である。Dが実行できるのは、A、B、Cが実行を終了した後である。つまり、Aの実行終了までの130ミリ秒(180-50)を終り、Bの実行終了まで80ミリ秒、Cの実行終了まで40ミリ秒待って、実行(20ミリ秒)を開始したがいって、Dのターンアラウンドタイムは、 $130+80+40+20=270$ (ミリ秒)である。

前述から、到着順方式の場合の平均ターンアラウンドタイムは、
 $(180 + 250 + 270) \div 4 = 970 \div 4 = 242.5$ (ミリ秒)
 の (オ) である。

空欄b:ラウンド

0: ラウンドロビン方式の場合のターンアラウンドタイム平均値を求める。

本問の場合、タイムクワウンダムは20ミリ秒ごとに実行を中断し、待ち行列の最後尾に再登録する。この方式では、待ち行列のプロセスの並びも把握していないかもしれない。タイムテーブルを描くと次のようになる。先ほど同様、これはあくまで例である。

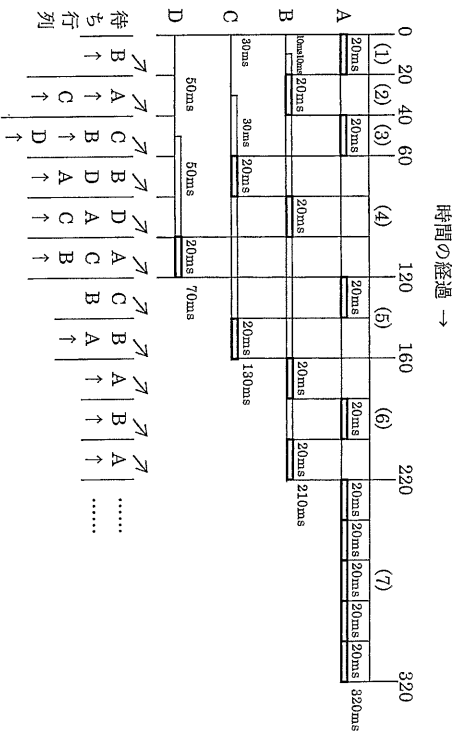


図2 ラウンドロビン方式のタイムテーブル例

[タイムテーブルの解説]

(1) 部分：最初の到着したAが待ち行列に登録され、すぐに取り出されて実行を開始する。(前述のように登録、取出しに要する時間は無視してよい) Aのターンアラウンドタイムが始まる。Aの開始から10ミリ秒後にBが到着して待ち行列に登録される。この時点からBのターンアラウンドタイムが始まる。

(2) 部分：Aの実行が中断されて待ち行列に再登録される。待ち行列の先頭にあったBが取り出されて実行を開始する。10ミリ秒後（Aの到着から30ミリ秒後）、Cが到着して待ち行列に登録される（先頭からA、Cと登録されている）。この時点からCのターンアラウンドタイムが始まる。

(3) 部分: Bの実行が中断されて待ち行列に再登録される。待ち行列の先頭はAなので、取り出してAを実行する。その10ミリ秒後(最初のAの実行開始から50ミリ秒後)、Dが到着して待ち行列に登録される(先頭からC、B、Dと登録されている)。この時点からDのターンラウンドタイムが始まる。

(4) 部分: その後も同様に中断されたプロセスは待ち行列に再登録され、先頭から取り出して実行する。3回分のタイムクォータムの間にC、B、Dといふ順で実行される。Dの処理時間は20ミリ秒であり、1回のタイムクォータムで終了となるので、待ち行列に再登録されることはない。Dのターンラウンドタイムは、待ち時間と実行時間で $50+20=70$ (ミリ秒)である。D終了時点で、待ち行列には先頭からA、C、Bと登録されている。

(5) 部分：待ち行列から A が取り出され実行される。20 ミリ秒で中断され、待ち行列に再登録される。次に C を取り出し実行する。C の処理時間は 4 ミリ秒であり、このタスクがオンタイムで終了となるので、待ち行列に再度登録されることはない。C のタスクがオンタイムで終了し、待ち行列と実行時間を合わせて $30 + 20 + 60 + 20 = 130$ (ミリ秒) である。待ち行列には、先頭から B, A が登録されている。

(6) 部分：取出し，実行，中断，再登録と繰り返して，B, A, Bと実行したところで，Bは終了する。Bのターンアラウンドタイムは，待ち時間と実行時間を合わせて $10+20+40+20+60+20+20+20=210$ (ミリ秒)である。待行列にはAだけが登録されている。

(7) 部分: Aが終了するまで取出し, 実行, 中断, 再登録と繰り返す。Aの
 一ソフトウェアプログラムのAは, 待ち時間と実行時間を含わせて, $20+20+20+6$
 $+20+40+20+20+20 \times 5 = 320$ (ミリ秒)である。

以上のことから、求めるターンアラウンドタイム平均値は
 $(70+130+210+320) \div 4 = 730 \div 4 = 182.5$ (ミリ秒)

の(ウ)である。

このように、同じプロセス群であり、すべての処理が終了するのに要する時間ともに 320 ミリ秒 (CPU を休ませることなく使用) であるが、ターンアラウンドタイムの平均値は方式によって、大きな差ができることが分かる。

【雜考】

設問1] a-オ, b-ウ

【設問2】 c-ウ, d-オ

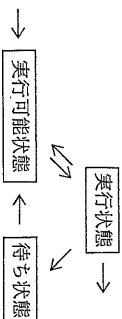
【解説】

CPUの割当て方式に関する問題である。この題材は、午前問題の出題範囲に含まれているので、各方式について理解しているはずである。しかし、たとえば理解に自信がなくても、説明に沿って考えていけば解ける問題である。この際、確実に理解しておくとよいだろう。

その方式として、まず、到着順方式とラウンジ方式の説明がされている。

到着順方式は、文字通り到着順に実行していく方式である。到着順にプロセスを実行の待ち行列の末尾に登録し、実行中のプロセスが終了したら、待ち行列の先頭からプロセスを一つ取り出して実行を開始するという手順である。一つのプロセスの実行が始まったら、そのプロセスが終了するまで CPU を割り当てられるので、処理時間の長いプロセスがあると、後続のプロセスは待たされることになる。

ところで、問題の説明中に、「待ち行列に登録されているプロセスの状態を実行可能状態、実行中のプロセスの状態を実行状態と呼ぶ」とある。「実行可能状態」、「実行状態」に加え、プロセスの出力要求を出して、その入出力処理の終了待ちの状態のことである。プロセスが入出力要求を出した場合、この「待ち状態」は考慮しなくてよい。「実行可能状態」と「実行状態」の行き来だけを考えればよい。



ラウンドロビン方式は、幾つものプロセスが、一定時間（タイムクォンタム）ずつ、いわば輪番制で CPU を割り当てられる方式である。到着順にプロセスを実行の待ち行列の末尾に登録するのは、到着順方式と同じである。プロセスの実行は、待ち行列の先頭から一つ取り出し開始するが、タイムクォンタムだけ経過すると、実執行を中断し、そのプロセスは待ち行列の末尾に再登録する。次は、待ち行列の先頭からプロセスを一つ取り出し実行を開始するのである。タイムクォンタム以内で終了すれば、もう待ち行列に再登録する必要はない。この方式では、遅れてきたプロセスでも、必ず実行の順番が回ってくる。

このような実行順序の決定方式の効果を示す指標として、ターンアラウンドタイムがある。ターンアラウンドタイムは、プロセスが待ち行列に到着してから実行が終了するまでの時間である。

[設問1]

ここでは、表1に示される四つのプロセスについて、両方式で実行した場合のタスクランタイムの平均値が問われている。平均値ということで、方式ごとに四つすべてのターンプラウランドタイムを調べることになる。頭の中だけで考えて、解答かき写すのならばそれにすることはない。しかし、このような問題では、自分自身からオペレーティングシステムになったつもりで、各方式の手順を守りつつ、タイムテーブルのような図を描いて考えとよい。プロセスの状態（実行・実行可能状態）、待ち行列の並びなどが、はつきりしてくるだろう。試験会場では方眼用紙も定規もないが、大ざっぱであればよいので、線を引いて、実行過程、実行時間など必要と思われる情報をメモ書きしていくとよい。

なお、設問中に「プロセスの取出し、及び中断の処理でのオーバヘッドは考えない」とあるのは、実行に要する時間だけ考えればよいことである。実際には、待ち行列に関して取出しや再登録、プロセスごとの情報退避や再設定などに時間を要するはずであるが、問題を解く上では、それらを考慮しなくてよいことである。また、「CPUを割り当てられたプロセスは、タイムクォンタム以外で中断することはない」とあるのは、例えば、データの入出力等で実行を中断することはないということである。これらの注意書きは、問題を簡略化して考えればよいという配慮である。

空欄 a : 到着順方式の場合のターニングラウンドタイム平均値を求める。

四つのプロセスの待ち行列への到着順は、A、A の 10 ミリ秒後に B、A の 30 ミリ秒後に C、A の 50 ミリ秒後に D、の順である。待ち行列に登録された時刻から、ターニングラウンドタイムを計算することになる。タイムテーブルを描くと、次のようになる。これは、あくまでタイムテーブルの例であって自分なりの描き方だよ。

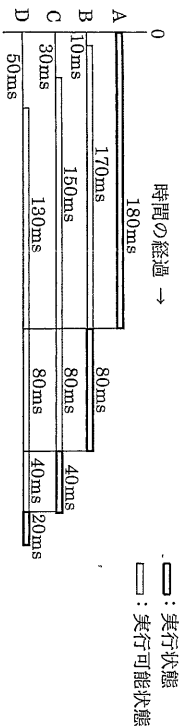


図 1 到着順方式のタイムテーブル例

[タイムテーブルの解説]

A:「Aが到着したとき、実行可能状態及び実行状態のプロセスはない」ということから、Aが待ち行列に到着するとすぐに実行できることになる。したがって、Aのターンアラウンドタイムは、実行時間の180ミリ秒だけである。B: Aが実行を開始して10ミリ秒後に到着する。そのときは、まだAの実行中である。Bが実行できるのは、Aの実行が終了してからである。つまり、IはAの実行(180ミリ秒)が終わるまで、待ち行列で170(180-10)ミリ秒実行を待ち、その後実行(80ミリ秒)を開始する。したがって、Bのターンアラウンドタイムは、 $170+80=250$ (ミリ秒)である。

C: Aが実行を開始して30ミリ秒後に到着する。そのときは、まだAは実行中である。Cが実行できるのは、Aの次にBが実行を終了した後である。つまり、Aの実行終了までの150(180-30)ミリ秒、Bの実行終了まで80ミリ秒待つ、実行(40ミリ秒)を開始する。したがって、Cのターンアラウンドタイムは、150+80+40=270(ミリ秒)である。

D: Aが実行を開始して50ミリ秒後に到着する。そのときは、まだAは実行中である。Dが実行できるのは、A、B、Cが実行を終了した後である。つまり、Aの実行終了までの130ミリ秒(180-50)を終り、Bの実行終了まで80ミリ秒、Cの実行終了まで40ミリ秒待って、実行(20ミリ秒)を開始したがいって、Dのターンアラウンドタイムは、 $130+80+40+20=270$ (ミリ秒)である。

前述から、到着順方式の場合の平均ターンアラウンドタイムは
 $(180 + 250 + 270 + 270) \div 4 = 970 \div 4 = 242.5$ (ミリ秒)
 の(オ)である。

空欄b:ラウンド

b: ラウンドロビン方式の場合のターンアラウンドタイム平均値を求める。

本間の場合、タイムクォンタムは20ミリ秒である。20ミリ秒ごとに実行を中断し、待ち行列の最後尾に再登録する。この方式では、待ち行列のプロセスの並びも把握していないなければならない。タイムテーブルを描くようになる。先ほど同様、これはあくまで例である。

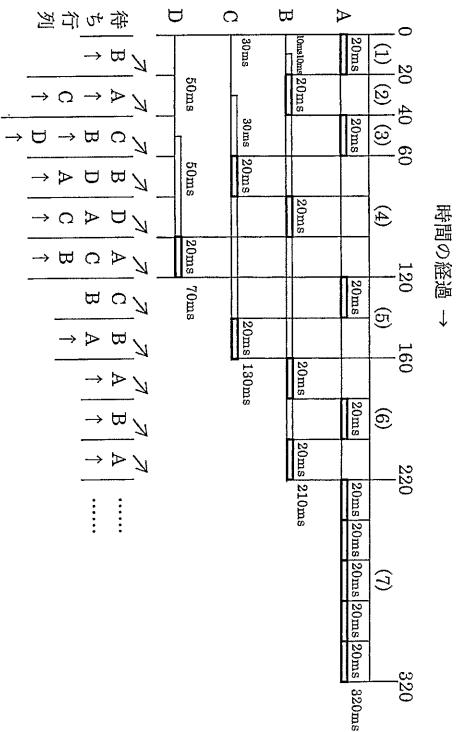


図2 ラウンドロビン方式のタイムテーブル例

[タイムテーブルの解説]

(1) 部分：最初の到着したAが待ち行列に登録され、すぐに取り出されて実行を開始する。(前述のように登録、取出しに要する時間は無視してよい) Aのターンアラウンドタイムが始まる。Aの開始から10ミリ秒後にBが到着して待ち行列に登録される。この時点からBのターンアラウンドタイムが始まる。

(2) 部分：Aの実行が中断されて待ち行列に再登録される。待ち行列の先頭にあったBが取り出されて実行を開始する。10ミリ秒後（Aの到着から30ミリ秒後）、Cが到着して待ち行列に登録される（先頭からA、Cと登録されている）。この時点からCのターンアラウンドタイムが始まる。

(3) 部分: Bの実行が中断されて待ち行列に再登録される。待ち行列の先頭はAなので、取り出してAを実行する。その10ミリ秒後(最初のAの実行開始から50ミリ秒後)、Dが到着して待ち行列に登録される(先頭からC、B、Dと登録されている)。この時点からDのターンアラウンドタイムが始まる。

(4) 部分: その後も同様に中断されたプロセスは待ち行列に再登録され、先頭から取り出して実行する。3回分のタイムクォータムの間にC、B、Dといふ順で実行される。Dの処理時間は20ミリ秒であり、1回のタイムクォータムで終了となるので、待ち行列に再登録されることはない。Dのターンアラウンドタイムは、待ち時間と実行時間で $50+20=70$ (ミリ秒)である。D終了時点で、待ち行列には先頭からA、C、Bと登録されている。

(5) 部分：待ち行列から A が取り出され実行される。20 ミリ秒で中断され、待ち行列に再登録される。次に C を取り出し実行する。C の処理時間は 4 ミリ秒であり、このタスクがオンタイムで終了となるので、待ち行列に再度登録されることはない。C のタスクがオンタイムで終了し、待ち行列と実行時間を合わせて $30 + 20 + 60 + 20 = 130$ (ミリ秒) である。待ち行列には、先頭から B, A が登録されている。

(6) 部分：取出し，実行，中断，再登録と繰り返して，B, A, Bと実行したところで，Bは終了する。Bのターンアラウンドタイムは，待ち時間と実行時間を合わせて $10+20+40+20+60+20+20+20=210$ (ミリ秒)である。待行列にはAだけが登録されている。

(7) 部分: Aが終了するまで取出し, 実行, 中断, 再登録と繰り返す。Aの
 一ソフトウェアプログラムのAは, 待ち時間と実行時間を含わせて, $20+20+20+6$
 $+20+40+20+20+20 \times 5 = 320$ (ミリ秒)である。

以上のことから、求めるターンアラウンドタイム平均値は
 $(70+130+210+320) \div 4 = 730 \div 4 = 182.5$ (ミリ秒)

の(ウ)である。

このように、同じプロセス群であり、すべての処理が終了するのに要する時間ともに 320 ミリ秒 (CPU を休ませることなく使用) であるが、ターンアラウンドタイムの平均値は方式によって、大きな差ができることが分かる。