

表 1 記号の説明

記号	ビット数	内容
op	8	00h~FFh で示されるいずれかの命令コードが指定される。
$r_1$	2	0~3 で示されるいずれかのレジスタ番号が指定される。
$r_2$	2	0~3 で示されるいずれかのレジスタ番号が指定される。
X	2	指標レジスタ修飾を行うときは、指標レジスタ修飾に使用するレジスタを表す 1~3 で示されるいずれかのレジスタ番号が指定される。 指標レジスタ修飾を行わないときは、0 が指定される。
D	1	1 語命令のときは、1 が指定される。 2 語命令のときは、0 が指定される。
I	1	間接アドレス指定を行うときは、1 が指定される。 間接アドレス指定を行わないときは、0 が指定される。
adr	16	0~999 で示されるいずれかの値（番地を表す値）が指定される。

表 2 実効アドレスの算出方法

D	I	実効アドレス
0	0	adr+[X で指定されたレジスタ]
0	1	[adr+[X で指定されたレジスタ]]
1	1	[ $r_2$ で指定されたレジスタ]

注 []は、[]内のレジスタ又は番地に格納されている内容を示す。

表 3 ソースレジスタの指定方法

D	I	ソースレジスタ
1	0	$r_2$ で指定されたレジスタ

表 4 命令の仕様（一部）

命令コード	動作
10h	実効アドレスに格納されている内容又はソースレジスタの内容を、 $r_1$ で指定されたレジスタに足し込む。
20h	実効アドレスに格納されている内容又はソースレジスタの内容を、 $r_1$ で指定されたレジスタに設定する。
30h	$r_1$ で指定されたレジスタの内容を、実効アドレスに格納する。
FFh	プログラムを終了する。

設問1 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。

命令を実行する前のレジスタの内容は，図2のとおりとする。この状態で図3のプログラムを先頭の700番地から順に実行した。

命令語2の実行が終わった時点での，レジスタ番号1のレジスタの内容は  a  である。命令語5の実行が終わった時点での，レジスタ番号1のレジスタの内容は  b  であり，レジスタ番号3のレジスタの内容は  c  である。

レジスタ番号	内容
0	100
1	200
2	300
3	400

図2 レジスタの内容

主記憶番地	内容	
700	2042h	命令語1
701	10D2h	
702	3090h	命令語3
703	0000h	
704	1013h	命令語4
705	2042h	命令語5
706	FF02h	命令語6

図3 プログラム

解答群

ア 100

イ 200

ウ 300

エ 400

オ 500

カ 600

設問2 レジスタと主記憶の内容が図4に示す値のとき、レジスタ番号1のレジスタに100を設定する命令語の記述として誤りであるものを、解答群の中から選べ。

レジスタ番号	内容
2	100
3	101

主記憶番地	内容
100	101
101	100

図4 レジスタと主記憶の内容

解答群

	1 語目					2 語目
	op	$r_1$	$r_2$ 又は X	D	I	adr
ア	20h	1	2	0	1	0000h
イ	20h	1	2	1	0	
ウ	20h	1	3	0	0	0000h
エ	20h	1	3	0	1	0000h
オ	20h	1	3	1	1	

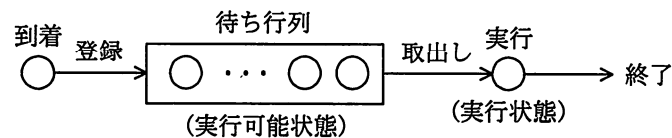
問2 CPUの割当て方式に関する次の記述を読んで、設問1, 2に答えよ。

オペレーティングシステムの役割の一つとして、プロセスにCPUを割り当てること  
がある。そして、プロセスの実行順序を決定する方式には、次のようなものがある。

(1) 到着順方式

到着順にプロセスを待ち行列の末尾に登録する。実行中のプロセスが終了する  
と、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。

到着順方式を図1に示す。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。



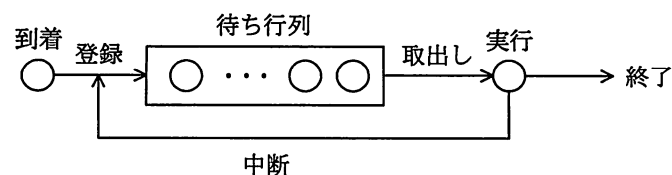
注 ○はプロセスを表す。

図1 到着順方式

(2) ラウンドロビン方式

到着順にプロセスを待ち行列の末尾に登録する。実行中のプロセスが終了する  
と、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。また、実行  
中のプロセスが一定時間（以下、タイムクウォンタムという）を経過したら、実  
行を中断して、待ち行列の末尾に再登録し、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。

ラウンドロビン方式を図2に示す。



注 ○はプロセスを表す。

図2 ラウンドロビン方式

これらの方式の効率を示す指標としてターンアラウンドタイムがある。ここで、ターンアラウンドタイムとは、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り、プロセスの実行順序に影響される。

なお、このコンピュータシステムの CPU は一つであり、CPU は同時に一つのプロセスしか実行できない。

設問 1 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。

四つのプロセス A～D があり、各プロセスの到着時刻と処理時間を表 1 に示す。表 1 において、到着時刻とは、プロセス A が待ち行列に到着した時刻を 0 としたときの各プロセスが到着する時刻であり、処理時間とは、各プロセスの処理が完了するために必要な CPU の処理時間である。

表 1 プロセスの到着時刻と処理時間

プロセス	到着時刻 (ミリ秒)	処理時間 (ミリ秒)
A	0	180
B	10	80
C	30	40
D	50	20

このとき、到着順方式におけるターンアラウンドタイムの平均は  a  ミリ秒である。そして、タイムクォンタムが 20 ミリ秒のとき、ラウンドロビン方式におけるターンアラウンドタイムの平均は  b  ミリ秒である。ここで、プロセス 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ものとする。

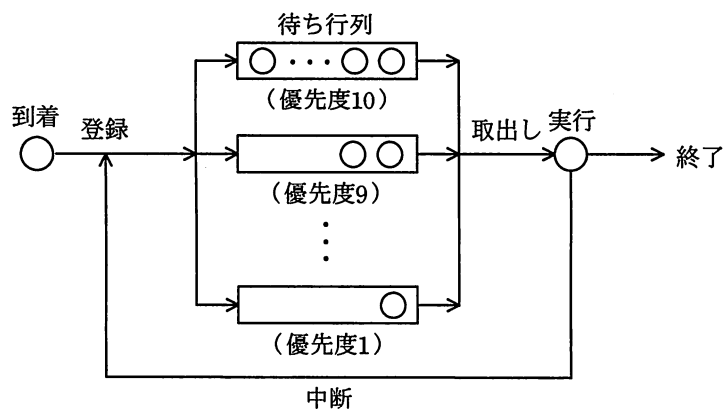
なお、プロセスの登録と取出し、及び中断の処理でのオーバヘッドは考えない。また、CPU を割り当てられたプロセスは、タイムクォンタム以外で中断することはない。

解答群

- |         |         |         |
|---------|---------|---------|
| ア 80.0  | イ 102.5 | ウ 182.5 |
| エ 192.5 | オ 242.5 |         |

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。

プロセスの実行順序を決める別の方式に優先度順方式がある。優先度順方式の例を図 3 に示す。プロセスにはあらかじめ優先度が付けてあり，待ち行列は優先度ごとに用意してある。ここで，優先度は 1 ～ 10 の 10 種類で，値の大きい方が優先度は高い。



注 ○ はプロセスを表す。

図 3 優先度順方式の例

この方式では，次のとおりにプロセスの実行を制御する。

- ① プロセスを優先度に対応した待ち行列の末尾に登録する。
- ② プロセスが登録されている優先度の最も高い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。
- ③ 実行中のプロセスの優先度が 2 以上のとき，実行時間が 20 ミリ秒経過するごとに優先度を一つ下げる。優先度を下げた結果，実行中のプロセスの優先度が実行可能状態にある優先度の最も高いプロセスよりも低くなった場合，実行中のプロセスを中断して，①に戻る。
- ④ 実行中のプロセスが終了した場合，②に戻る。

優先度順方式において，あるプロセスが終了した時点で表 2 に示す三つのプロセスだけが優先度に対応した待ち行列に登録されていたとする。このとき，三つのプロセスが終了する順番は  c  である。そして，プロセス

B の実行が終了したときのプロセス B の優先度は d である。ここで、三つのプロセスが終了するまで新たに到着するプロセスはないものとする。

なお、プロセスの登録と取出し、及び中断の処理でのオーバーヘッドは考えない。また、CPU を割り当てられたプロセスは、タイムクウォンタム以外で中断することはない。

表 2 プロセスの処理時間と優先度の初期状態

プロセス	処理時間 (ミリ秒)	優先度
A	60	6
B	70	8
C	100	5

cに関する解答群

- ア A, B, C      イ A, C, B      ウ B, A, C      エ B, C, A  
 オ C, A, B      カ C, B, A

dに関する解答群

- ア 1      イ 2      ウ 3      エ 4  
 オ 5      カ 6