

# 平成27年度後期 情報検定

<実施 平成28年2月14日（日）>

## 基本スキル

(説明時間 13:00~13:10)

(試験時間 13:10~14:10)

- ・試験問題は試験開始の合図があるまで開かないでください。
- ・解答用紙（マークシート）への必要事項の記入は、試験開始の合図と同時に行いますので、それまで伏せておいてください。
- ・試験開始の合図の後、次のページを開いてください。＜受験上の注意＞が記載されています。必ず目を通してから解答を始めてください。
- ・試験問題は、すべてマークシート方式です。正解と思われるものを1つ選び、解答欄の○をHBの黒鉛筆でぬりつぶしてください。2つ以上ぬりつぶすと、不正解になります。
- ・辞書、参考書類の使用および筆記用具の貸し借りは一切禁止です。
- ・電卓の使用が認められます。ただし、下記の機種については使用が認められません。

### <使用を認めない電卓>

1. 電池式（太陽電池を含む）以外の電卓
2. 文字表示領域が複数行ある電卓（計算状態表示の一行は含まない）
3. プログラムを組み込む機能がある電卓
4. 電卓が主たる機能ではないもの
  - \* パソコン（電子メール専用機等を含む）、携帯電話（PHS）、スマートフォン、タブレット、電子手帳、電子メモ、電子辞書、翻訳機能付き電卓、音声応答のある電卓、電卓付腕時計等
5. その他試験監督者が不適切と認めるもの

## ＜受験上の注意＞

1. この試験問題は15ページあります。ページ数を確認してください。  
乱丁等がある場合は、手をあげて試験監督者に合図してください。  
※問題を読みやすくするために空白ページを設けている場合があります。
2. 解答用紙（マークシート）に、受験者氏名・受験番号を記入し、受験番号下欄の数字をぬりつぶしてください。正しく記入されていない場合は、採点されませんので十分注意してください。
3. 試験問題についての質問には、一切答えられません。自分で判断して解答してください。
4. 試験中の筆記用具の貸し借りは一切禁止します。筆記用具が破損等により使用不能となった場合は、手をあげて試験監督者に合図してください。
5. 試験を開始してから30分以内は途中退出できません。30分経過後退出する場合は、もう一度、受験番号・マーク・氏名が記載されているか確認して退出してください。なお、試験終了5分前の合図以降は退出できません。試験問題は各自お持ち帰りください。
6. 試験後にお知らせする合否結果（合否通知）、および合格者に交付する「合格証・認定証」はすべて、Webページ（PC、モバイル）での認証によるデジタル「合否通知」、デジタル「合格証・認定証」に移行しました。
  - ①団体宛にはこれまでと同様に合否結果一覧ほか、試験結果資料一式を送付します。
  - ②合否等の結果についての電話・手紙等でのお問い合わせには、一切応じられませんので、ご了承ください。

問題を読みやすくするために、  
このページは空白にしてあります。

問題 1 次のプロジェクトの進捗・コスト管理に関する各設問に答えよ。

<設問 1> 次の PERT に関する記述中の  に入れるべき適切な字句を解答群から選べ。

プロジェクトにおける作業の関連性や日程を管理するための手法として PERT がある。PERT では、図 1 に示すようなアローダイアグラムで関連を図示し、矢印は作業を、丸印は結合点（ノード）を表す。矢印には作業名やその作業の標準の作業日数が書き込まれる。

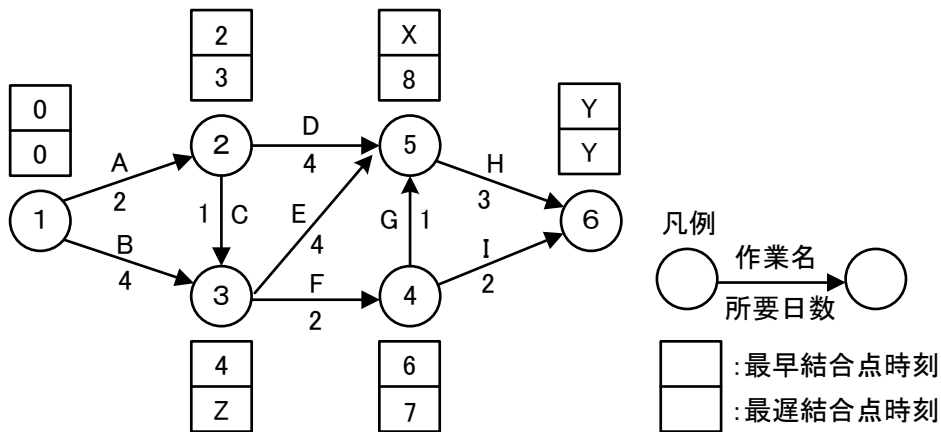


図 1 PERT の例

前提条件として、結合点に到達するすべての作業が終了してから、次の作業を開始できる。

最早結合点時刻とは、結合点から始まる作業を最も早く開始できる時刻である。複数の作業が到達する場合は最も大きい値が選択される。なお、最終結合点の最早結合点時刻が、このプロジェクトの総所要日数である。

最遅結合点時刻とは、プロジェクトの総所要日数に影響を与えずに、この結合点から始まる作業を最も遅く開始できる時刻である。複数の作業が開始される場合は最も小さい値が選択される。

また、所要時間が最も長い経路をクリティカルパスという。したがってクリティカルパスは、余裕のない作業を結んだ経路になる。

PERT 図の作成は、まず、最初の結合点から最終の結合点まで順に最早結合点時刻だけを求めていく。次に、最終結合点の最遅結合点時刻にプロジェクトの総所要日数を記入し、最初の結合点まで逆に戻りながら各結合点の最遅結合点時刻を求めていく。

図 1 において、結合点⑤の最早結合点時刻(図 1 の X)は  (1) であり、プロジェクトの総所要日数(図 1 の Y)は  (2) である。また、結合点③の最遅結合点時刻(図 1 の Z)は  (3) である。さらに、このプロジェクトのクリティカルパスは  (4) である。

(1) ~ (3) の解答群

- |      |      |       |       |
|------|------|-------|-------|
| ア. 3 | イ. 4 | ウ. 6  | エ. 7  |
| オ. 8 | カ. 9 | キ. 10 | ク. 11 |

(4) の解答群

- |                  |                  |
|------------------|------------------|
| ア. A → C → E → H | イ. A → C → F → I |
| ウ. B → E → H     | エ. B → F → I     |

<設問 2 > 次の EVM に関する記述中の  に入れるべき適切な字句を解答群から選べ。

EVM(Earned Value Management)とは、プロジェクトの進捗とコストの両面で計画(予定)と実績を比較して管理する手法である。

EVM では、計画段階でスケジュールに対応させた計画価値(PV : Planned Value)を作成し、測定時点でどれだけの価値を作り上げることができているかという達成価値(EV : Earned Value)と比較することで、プロジェクトの実績を把握する。価値を表す尺度として通常は金額を用いる。

測定時点(現時点)での達成価値と、計画していた価値との差(PV-EV)をスケジュール差異といい、比率(EV÷PV)をスケジュール効率指数という。また、現時点での達成価値を計画していた時間と、現時点の時間の差が遅れの時間となる

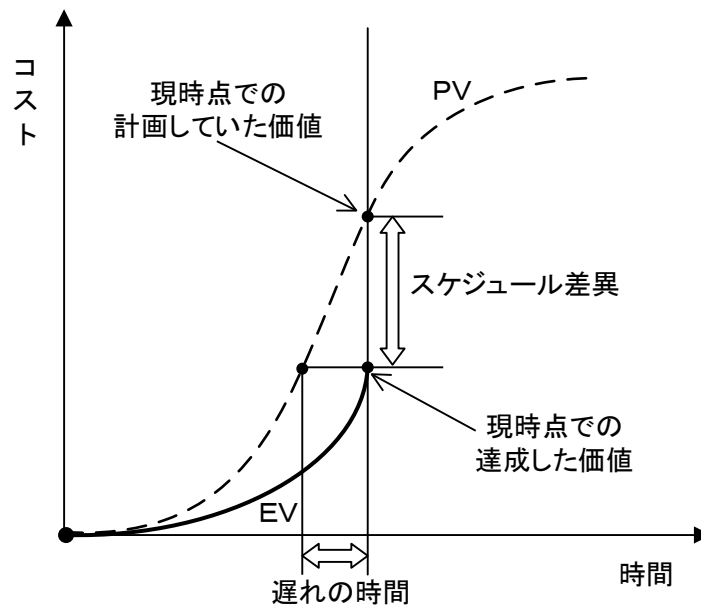


図 2 EVM の図

このプロジェクト完了時点の価値を 9000 万円とし、プロジェクトの期間を 300 日としたとき、30 日ごとの計画価値 (PV) を表に示す。180 日時点の達成価値が 3600 万円であるとき、スケジュール差異は  万円であり、スケジュール効率指数は  である。また、遅れの時間は  日である。

表 プロジェクトの計画価値 (PV)

経過日数(日)	30	60	90	120	150	180	210	240	270	300
計画価値(万円)	200	500	1000	2000	3600	6000	8000	8500	8800	9000

(5) の解答群

ア. 2000                      イ. 2400                      ウ. 3600                      エ. 5400

(6) の解答群

ア. 0.2                      イ. 0.4                      ウ. 0.6                      エ. 0.8

(7) の解答群

ア. 30                      イ. 60                      ウ. 90                      エ. 120

問題を読みやすくするために、  
このページは空白にしてあります。

問題2 次の文字データとその表現に関する記述を読み、各設問に答えよ。

コンピュータでは、特定のビットパターンを割り当てることで文字を表現する。半角の英数字や記号は1バイトを用い、漢字やひらがな等の全角文字の場合は2バイト以上を用いる。このように表現されたものを文字コードと呼ぶ。文字コードには、利用したい文字の集まりである文字集合と、どの文字集合に合わせて変換するかを決定するエンコードの概念が含まれている。

<設問1> 次の1バイト文字データに関する記述中の  に入れるべき適切な字句を解答群から選べ。

現在使われている1バイトの文字コードはASCIIコードが世界的に利用されている。これは、ANSI（米国標準規格協会）で制定された7ビットを基にしており、8ビット目はデータ通信に用いるときのパリティビットとしている。

8ビットで表現できる文字コードは256種類だが、その半分はメーカーや各国の標準化団体が独自の文字を割り当てている。日本では、ASCIIコードを基にし、半角のカタカナを収録したJIS X 0201（ANKコード）が使われている。

次の図は、ASCIIコードに収録されている文字とコードを対応させたものである。なお、 $(000\ 0000)_2 \sim (001\ 1111)_2$ （16進数の00~1F）、および $(111\ 1111)_2$ （16進数の7F）は制御文字である。

b7	b6	b5	b4	b3	b2	b1	列	0	1	2	3	4	5	6	7
							行	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	NUL	DLE	空白	0	@	P	`	p
0	0	0	0	1	1	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	0	0	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	0	0	0	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	0	0	0	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	0	0	0	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	0	0	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	0	0	0	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	0	0	0	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	0	0	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	0	0	0	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	0	0	0	12	FF	FS	,	<	L	¥	l	
1	1	0	1	0	0	0	13	CR	GS	-	=	M	]	m	}
1	1	1	0	0	0	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	0	0	0	15	SI	US	/	?	0	_	o	DEL

図 ASCIIコード



この図から、「Z」が割り当てられているコードは $(101\ 1010)_2$  (16進数の5A)であることがわかる。同様に、「A」、「a」、「3」、「#」という文字は、それぞれ16進数で次のようなコードになる。

A	…	<input type="text" value="(1)"/>
a	…	<input type="text" value="(2)"/>
3	…	<input type="text" value="(3)"/>
#	…	<input type="text" value="(4)"/>

また、コードが16進数で順に57, 33, 43となる文字列はである。

**(1) ~ (4) の解答群**

- |       |       |       |       |
|-------|-------|-------|-------|
| ア. 14 | イ. 16 | ウ. 23 | エ. 32 |
| オ. 33 | カ. 41 | キ. 61 |       |

**(5) の解答群**

- |        |        |        |        |
|--------|--------|--------|--------|
| ア. u34 | イ. u3C | ウ. W34 | エ. W3C |
|--------|--------|--------|--------|

<設問2> 次の日本語の文字コードに関する記述に関連の深い字句を解答群から選べ。

- (6) Windows などのパソコン上で標準的に用いられる2バイトコードである。1バイト文字と2バイト文字の区別は先頭の1バイトで判断できるようになっており、シフトイン、シフトアウトのような制御文字を使用しない。
- (7) AT&T が日本語 UNIX システム諮問委員会の提案に基づいて、文字コードの各国定義部分に日本語の文字集合を割り当てたもので、UNIX 上で日本語を利用するため用いられる。
- (8) 米国の IT 関連企業が中心となって提唱して標準化された文字コードで、1つの文字コード体系で各国の言語を処理するものである。全ての文字を2バイトで表現するため半角文字や全角文字を統一的に利用できる。現在は様々な文字が追加され、一部では3バイト以上で定義されている。

**(6) ~ (8) の解答群**

- |              |             |            |
|--------------|-------------|------------|
| ア. EBCDIC    | イ. EUC-JP   | ウ. JIS     |
| エ. Shift JIS | オ. TRON コード | カ. Unicode |

問題3 次の論理演算に関する説明を読み、各設問に答えよ。

論理演算とは、真と偽や1と0のように、2つの値のいずれか一方の値を持つデータ間で行われるもので、結果も真と偽や1と0となる。論理演算を表にまとめたものを真理値表と呼ぶ。また、次の手順により、真理値表から論理式を求めることができる。

[手順1] 出力が“1”の行に注目し、入力が“1”の項はそのまま、入力が“0”の項は否定の形式にして、各項の論理積をとる。

次の真理値表から論理式を求めてみる。なお、論理和を“+”，論理積を“ $\cdot$ ”，Aの否定を $\bar{A}$ で表す。

入力		出力	
A	B	X	
0	0	0	
0	1	1	$\rightarrow \bar{A} \cdot B$
1	0	1	$\rightarrow A \cdot \bar{B}$
1	1	1	$\rightarrow A \cdot B$

図1 真理値表

[手順2] 手順1で作った項の論理和をとる。

$$\bar{A} \cdot B + A \cdot \bar{B} + A \cdot B$$

[手順3] 論理法則 ( $A + A = A$  など) を利用し、簡素化する。

$$\begin{aligned} \bar{A} \cdot B + A \cdot \bar{B} + A \cdot B &= B \cdot (\bar{A} + A) + A \cdot (\bar{B} + B) \\ &= B \cdot 1 + A \cdot 1 \\ &= B + A \end{aligned}$$

となり、AとBの論理和を表していることがわかる。

<設問 1 > 次の真理値表から得られる論理式を解答群から選べ。

(1)																		
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><th colspan="2">入力</th><th>出力</th></tr> <tr><th>A</th><th>B</th><th>X</th></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	入力		出力	A	B	X	0	0	0	0	1	1	1	0	0	1	1	1
入力		出力																
A	B	X																
0	0	0																
0	1	1																
1	0	0																
1	1	1																

(2)																		
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><th colspan="2">入力</th><th>出力</th></tr> <tr><th>A</th><th>B</th><th>X</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	入力		出力	A	B	X	0	0	1	0	1	0	1	0	0	1	1	0
入力		出力																
A	B	X																
0	0	1																
0	1	0																
1	0	0																
1	1	0																

(3)																		
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><th colspan="2">入力</th><th>出力</th></tr> <tr><th>A</th><th>B</th><th>X</th></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	入力		出力	A	B	X	0	0	1	0	1	0	1	0	0	1	1	1
入力		出力																
A	B	X																
0	0	1																
0	1	0																
1	0	0																
1	1	1																

図 2 真理値表

(1) ~ (3) の解答群

- |                            |  |  |              |
|----------------------------|--|--|--------------|
| ア. $A$                     | イ. $\bar{A}$                           | ウ. $B$                                 | エ. $\bar{B}$ |
| オ. $\bar{A} \cdot \bar{B}$ | カ. $\bar{A} \cdot B + A \cdot \bar{B}$ | キ. $\bar{A} \cdot \bar{B} + A \cdot B$ |              |

<設問 2 > 次のビットの加算に関する記述中の   に入れるべき適切な字句を解答群から選べ。

同じけたにあるビット同士の加算を考える。1 ビットの加算は、図 3 のようになり、真理値表は表 1 のようになる。

A:	0	0	1	1
B:	<u>+ 0</u>	<u>+ 1</u>	<u>+ 0</u>	<u>+ 1</u>
	0 0	0 1	0 1	1 0

図 3 1 ビットの加算

表 1 1 ビットの加算結果表

		演算結果	
A	B	けた上がり (C)	同けたの和 (S)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

表 1 から、同けたの和 (S) は (4)、けた上がり (C) は (5) の論理式で表現できることがわかる。このように、二つの 2 進数を加算して同けたの値と

繰り上がり出力するものを (6) と呼ぶ。ただし、(6) は、下位からのけた上りを考慮していないため、最下位ビットの演算しか行えない。そこで、下位からのけた上りを含めた加算回路が必要である。これを (7) と呼ぶ。

(4) , (5) の解答群

ア.  $A \cdot B$

イ.  $A \cdot \bar{B}$

ウ.  $\bar{A} \cdot B$

エ.  $A + B$

オ.  $\bar{A} \cdot B + A \cdot \bar{B}$

(6) , (7) の解答群

ア. カウンタ

イ. シフトレジスタ

ウ. 全加算器

エ. 半加算器

オ. フリップフロップ

問題を読みやすくするために、  
このページは空白にしてあります。

問題 4 次の高速化に関する各設問に答えよ。

＜設問 1＞ 次のメモリアクセスの高速化に関する記述中の  に入れるべき適切な字句を解答群から選べ。

メモリは、大きく分けると揮発性の RAM と不揮発性の ROM に分けられる。

RAM には、主にキャッシュメモリに使用される  (1) とメインメモリ（主記憶装置）に使用される  (2) がある。 (1) は  (2) に比べると高速なので、CPU が読み出そうとするデータがキャッシュメモリに存在すればキャッシュメモリから、存在しなければメインメモリから呼び出すことで、メインメモリの平均アクセス時間の高速化を図ることができる。

データがキャッシュメモリにある確率をヒット率と呼ぶ。メインメモリへのアクセス時間が 70 n 秒、キャッシュメモリへのアクセス時間が 10 n 秒、ヒット率が 0.9 である場合の平均アクセス時間は  (3) n 秒となる。

また、メインメモリを複数のバンク（区画）に分け、連続するアドレスの内容を並列アクセスすることによって、アクセスの高速化を図る  (4) もある。

(1) , (2) , (4) の解答群

- ア. DRAM                      イ. SRAM                      ウ. ライトスルー  
エ. ライトバック              オ. メモリインターリーブ

(3) の解答群

- ア. 8                      イ. 16                      ウ. 64                      エ. 72

＜設問 2＞ 次の CPU の高速化技法に関する記述中の  に入れるべき適切な字句を解答群から選べ。

CPU はメインメモリに記憶されたプログラムから命令を取出し、実行する。例えば、この一連の動作を 6 ステージに分割した場合を考える。

- ① 命令の取出し
- ② 命令の解読
- ③ アドレス部の取出し
- ④ 実効アドレスの計算
- ⑤ データの取出し
- ⑥ 演算の実行

命令を逐次制御方式で実行する場合は、1命令ずつ各ステージを実行した後に次の命令を実行するので、演算装置や制御装置が動作しない時間が生じる。そのため、複数の命令を1ステージずつずらしながら同時に実行することで、処理を高速化する

**(5)**方式がある。

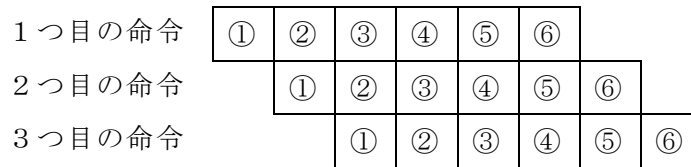


図 1 ステージずつずらして実行

**(5)**方式は、命令ごとの実行時間に大きな差があると高速化の効果は上がらない。そのため、命令セットアーキテクチャとして、命令の種類を減らし、できるだけ単純化して各命令の実行時間を均等にした**(6)**方式を採用している。しかし、分岐命令があると、後続の命令のステージの先読みが無駄になることや、前後の命令で同一データを使用するため、前の実行結果が格納されるまで次の計算ができないなど、**(7)**という乱れが発生すると効率が上がらない。

なお、各ステージの動作が  $10n$  秒で、分岐命令を含まない5個の命令を処理した場合、逐次制御方式では  $300n$  秒かかるが、**(5)**方式では**(8)** $n$  秒となり高速化される。

**(5) ~ (7) の解答群**

- ア. CISC
- イ. RISC
- ウ. スーパスカラ
- エ. パイプライン
- オ. パイプラインハザード

**(8) の解答群**

- ア. 50
- イ. 80
- ウ. 100
- エ. 150

問題5 次のオペレーティングシステムに関する各設問に答えよ。

＜設問1＞ 次のオペレーティングシステムの目的に関する記述中の□□□□に入れるべき適切な字句を解答群から選べ。

オペレーティングシステム(OS)の目的を分類すると、次の5つに集約できる。

① ハードウェア資源の有効活用

主記憶装置、入出力装置、補助記憶装置などのハードウェアをコンピュータシステム全体として効率よく利用する。

② □□(1)の短縮

無駄のない連続処理と操作性の向上、さまざまな処理形態への対応により、ジョブを投入してから結果が得られるまでの時間を短縮する。

③ □□(2)の向上

CPUやメモリなどを効率的に利用することにより、コンピュータが単位時間あたりに処理できる仕事量を増やす。

④ 応用ソフトウェアの負荷軽減

仮想記憶管理やファイルシステムによる入出力処理の効率化(データとプログラムの分離)によって実現する。

⑤ 信頼性と安全性の向上

RASまたはRASISと呼ばれる尺度で、システムの信頼性と安全性を評価して向上させる。RASISとは、故障が少なく安定して稼働する信頼性(Reliability)、必要なときにいつでも利用することができる可用性(Availability)、故障原因の発見や修理が容易にできる保守性(Serviceability)、システム内の情報を常に正しい状態に保つ保全性(Integrity)、正規の権限を持つ者のみが情報を利用できる機密性(Security)を指している。

(1)、(2)の解答群

ア. オーバヘッドタイム

イ. スループット

ウ. ターンアラウンドタイム

エ. ディスパッチ

＜設問2＞ 次のオペレーティングシステムの機能に関する記述中の□□□□に入れるべき適切な字句を解答群から選べ。

オペレーティングシステムの制御プログラムは、狭義のOSとも呼ばれ、さまざまな機能を持つプログラムの集合体である。その機能は次のように分類できる。

① □□(3)管理

□□(3)とは、利用者から見た仕事の単位である。□□(3)の投入から結果出力までの過程を、効率よく管理している。



② (4) 管理

(4) とは、OS から見た仕事の単位である。(4) (またはプロセス) に CPU を効率よく使うための割り当てを行う。割り当て方式の代表的なものとして、タイムスライスと呼ばれる一定時間を、できるだけ公平に割り当てる (5) がある。

③ データ管理／ファイル管理

ファイルシステムとも呼ばれ、データ操作を単純化する。ファイル編成方式とアクセス方式などが提供される。

④ 記憶管理

主記憶装置を効率よく利用するための機能で、実記憶管理と仮想記憶管理がある。

⑤ その他

障害管理、入出力管理、通信管理、セキュリティ管理、運用管理などがある。

(3) ～ (5) の解答群

ア. カーネル

イ. ジョブ

ウ. スプール

エ. タスク

オ. ページング

カ. ラウンドロビン

<設問 3 > 次のオペレーティングシステムに関する記述を読み、関係の深い字句を解答群から選べ。

(6) AT&T 社のベル研究所で、ミニコンピュータ用の OS として開発された。OS 自体が C 言語で記述されているため、C 言語との相性が良く C 言語とともに普及した。

(7) ヘルシンキ大学の Linus Torvalds によって作られた OS である。フリーソフトとして流通しているのもので無料で入手でき、ソースプログラムが公開されているので機能追加や修正が可能である。

(8) Microsoft 社が開発したパソコン用の OS である。GUI をベースにしており、ビジュアルな情報表示と、マウスの採用によって操作性が向上した。

(6) ～ (8) の解答群

ア. CP/M

イ. iOS

ウ. Linux

エ. Mac OS

オ. UNIX

カ. Windows

<メモ欄>

<メモ欄>

