

第1章

(1) 省略

(2) 省略

(3) 0~9            10種類  
A~Z(大文字) 26種類  
a~z(小文字) 26種類  
記号            50種類  
合計            112種類

したがって  $-\log_2\left(\frac{1}{112}\right) = 6.80735 \div 7\text{bit}$

(4) 省略

(5) ①  $70 \times 35 = 2450 \div 2.5\text{MB}$  (1ページ当たり)

②~⑥ 省略

(6) 省略

第2章

(1) ①  $1 + 2^{-1} + 2^{-3} + 2^{-4} = 1 + \frac{1}{2} + \frac{1}{8} + \frac{1}{16} = (1.6875)_{10}$

②  $1 - 2^{-1} + 2^{-3} = 1 + \frac{1}{2} + \frac{1}{8} = (1.625)_{10}$

③  $1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 32 + 16 + 4 + 2 = (54)_{10}$

④  $22 \div 2 = 11$  余り 0

$11 \div 2 = 5$  余り 1

$5 \div 2 = 2$  余り 1

$2 \div 2 = 1$  余り 0

$1 \div 2 = 0$  余り 1

したがって  $(22)_{10} = 10110$

⑤  $17 \div 2 = 8$  余り 1

$8 \div 2 = 4$  余り 0

$4 \div 2 = 2$  余り 0

$2 \div 2 = 1$  余り 0

$1 \div 2 = 0$  余り 1

したがって  $(17)_{10} = 10001$

⑥  $0.7 \times 2 = 1.4$  小数点を超える値 1

$0.4 \times 2 = 0.8$  小数点を超える値 0

$0.8 \times 2 = 1.6$  小数点を超える値 1

$0.6 \times 2 = 1.2$  小数点を超える値 1

$0.2 \times 2 = 0.4$  小数点を超える値 0

$0.4 \times 2 = 0.8$  小数点を超える値 0

$0.8 \times 2 = 1.6$  小数点を超える値 1

$0.6 \times 2 = 1.2$  小数点を超える値 1

$0.2 \times 2 = 0.4$  小数点を超える値 0

以下繰り返し(循環する)

したがって  $(0.7)_{10} = 0.1\overline{01100110}$

10進数で有限桁の数値も2進数表現では「循環小数」になることがある。

(2) (a)  $1.1011$

$+ 1.101$

$11.0101 \rightarrow 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-2} + 1 \times 2^{-4} = 2 + 1 + \frac{1}{4} + \frac{1}{8} = (3.3125)_{10}$

(b)  $10110$

$+ 10001$

$100111$

(c)  $10110$

$- 10001$

$101$

$$\begin{array}{r}
 \text{(d)} \quad \quad \quad 10110 \\
 \times \quad 10001 \\
 \hline
 \quad \quad 10110 \\
 \quad \quad 00000 \\
 \quad \quad 00000 \\
 \quad \quad 00000 \\
 \quad \quad 00000 \\
 + 10110 \\
 \hline
 \end{array}$$

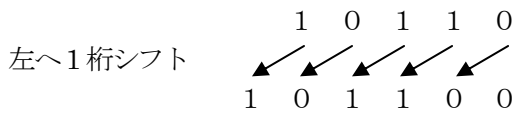
$$101110110 \rightarrow 2^8 + 2^6 + 2^5 + 2^4 + 2^2 + 2^1 = 256 + 64 + 32 + 16 + 4 + 2 = (374)_{10}$$

$$\begin{array}{r}
 \text{(e)} \quad 110110 \\
 \times \quad 10001 \\
 \hline
 \quad 110110 \\
 + 110110 \\
 \hline
 \end{array}$$

$$1110010110 \rightarrow 2^9 + 2^8 + 2^7 + 2^4 + 2^2 + 2^1 = 512 + 256 + 128 + 16 + 4 + 2 = (918)_{10}$$

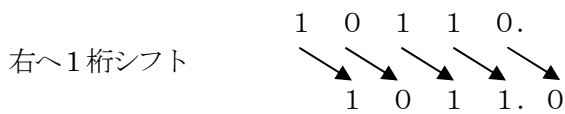
(3)  $(22)_{10} = 10110$  を 2 倍

$$(44)_{10} = 2^5 + 2^3 + 2^2 = 101100$$



$$(22)_{10} = 10110 \text{ を } \frac{1}{2} \text{ 倍}$$

$$(11)_{10} = 2^3 + 2^1 + 2^0 = 1011$$



(4) 例えば 8 ビットの数を

$$10101100 = (172)_{10}$$

とすると

$$\text{右へ 2 桁シフトした数 } a = 00101011 = (43)_{10} \dots\dots \frac{1}{4} \text{ 倍}$$

$$\text{左へ 1 桁シフトした数 } b = 101011000 = (344)_{10} \dots\dots 2 \text{ 倍}$$

右 2 桁シフトで  $\frac{1}{4}$  倍、左 1 桁シフトで 2 倍になる。

$$\text{したがって } b = a \times 8 \text{ (倍)}$$

- (5)  $255 \div 2 = 127$  余り 1  
 $127 \div 2 = 63$  余り 1  
 $63 \div 2 = 31$  余り 1  
 $31 \div 2 = 15$  余り 1  
 $15 \div 2 = 7$  余り 1  
 $7 \div 2 = 3$  余り 1  
 $3 \div 2 = 1$  余り 1  
 $1 \div 2 = 0$  余り 1

$$(255)_{10} = \underbrace{11111111}_{3 \text{ 7 7}}$$

$$8 \text{ 進数 } 3 \ 7 \ 7 \leftarrow 3 \times 8^2 + 7 \times 8^1 + 7 \times 8^0 = 192 + 56 + 7 = (255)_{10}$$

したがって  $(377)_8$  となる。また、

$$\underbrace{11111111}_{15 \ 15}$$

$$16 \text{ 進数 } F(=15) \ F(=15) \leftarrow 15 \times 16^1 + 15 \times 16^0 = 15 \times 16 + 15 = (255)_{10}$$

したがって  $(FF)_{16}$  となる。

- (6) それぞれ 2 進数で 5 桁(5 ビット)であるとして

$$(12)_{10} = 01100$$

$$(7)_{10} = 00111$$

となる。また、 $(7)_{10}$  の 2 の補数はビット列を反転して最後に 1 を加算すれば得られる。

したがって  $(-7)_{10} = 11000 + 1 = 11001$  となる。

加算

$$\begin{array}{r} \text{10 進表示} \quad \text{2 進表示} \\ 12 \quad 01100 \\ +) \quad 7 \quad 00111 \\ \hline 19 \leftarrow 10011 \end{array}$$

減算

$$\begin{array}{r} \text{10 進表示} \quad \text{2 進表示(補数)} \\ 12 \quad 01100 \\ +) \quad -7 \quad 11001 \\ \hline 5 \leftarrow 100101 \text{(先頭のビット削除)} \end{array}$$

$$\text{乗算 } 10 \text{ 進数 } (12)_{10} \times (7)_{10} = (84)_{10}$$

$$2 \text{ 進数 } 01100 \times 00111 = 0110000 + 011000 + 01100 = 1010100$$

- (7) 10 進数  $(24)_{10}$  は、2 進数 6 桁(6 ビット)であるとして

$$(24)_{10} = 01100$$

2 の補数  $(-24)_{10}$  はビット列を反転して最後に 1 を加算すれば得られる。

したがって  $(-24)_{10} = 10011 + 1 = 10100$  となる。

- (8)  $(22)_{10} = 010110$  であるから 2 の補数  $(-22)_{10}$  は

$$101001 + 1 = 101010$$

となる。

$$(17)_{10} + (-22)_{10} = 010001 + 101010 = 111011$$

111011 は 000101 の各桁を反転して 1 を加えた値となっている。

したがって  $000101 = (5)_{10}$  であるので、計算結果は  $(-5)_{10}$  である。

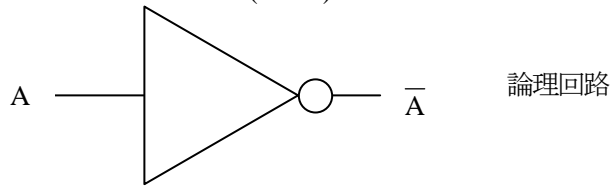
(9)

A	B	f
0	0	1
0	1	1
1	0	0
1	1	0

←  $\overline{A} \cdot \overline{B}$  論理最小項

←  $\overline{A} \cdot B$  論理最小項

$$f = \overline{A} \cdot \overline{B} + \overline{A} \cdot B = \overline{A} \cdot (\overline{B} + B) = \overline{A}$$



(10)  $(4A)_{16} = 1001010$  で、1 の個数は奇数(=3)となるので、1 の個数を偶数とするために上位に 1 を付加して、11001010 となる。

3章

- (1) 【ヒント】 WEB サーバ、Mail サーバ等の各種サーバが設置されていると考えられる。サーバの性能、OS はそれぞれ必要に応じて決められるので、大学により異なる結果になる。詳細はシステム管理部門に聞くこと。
- (2) 【ヒント】 「3-2-1 汎用コンピュータの歴史」を参考に第1世代～第4世代までまとめること。
- (3) 【ヒント】 「3-2-2 パーソナルコンピュータの歴史」を参考に米国、日本ごとに影響をまとめること。その際、デファクトスタンダードの概念(売れたものが標準)を考慮すること。

(4) 1)  $4 \text{億} = 4 \times 10^8$

正方形に敷き詰められているとすると、4億は  $4 \times 10^8 = (2 \times 10^4)^2$  だから、1辺(2m)に  $2 \times 10^4$  個の素子が並ぶことになる。  
 $2\text{m} = 2 \times 10^2 \text{cm} = 2 \times 10^3 \text{mm}$  に  $2 \times 10^4$  個だから、素子間の距離は、 $(2 \times 10^3 \text{mm}) \div (2 \times 10^4) = 0.1 \text{mm}$

2) 上記で1辺に  $2 \times 10^4$  個の素子が並ぶことがわかった。今回1個の単独のトランジスタの1辺が1cmであるから、CPUの1辺の長さは、以下ようになる。

$$1 \text{cm} \times 2 \times 10^4 = 2 \times 10^4 \text{cm} = 2 \times 10^2 \text{m} = 200 \text{m}$$

3)  $4 \text{億} = 4 \times 10^8$

1個をハンダ付けするのに10秒かかるから、全体を完成させるまでの時間は、 $4 \times 10^8 \times 10 \text{秒} = 4 \times 10^9 \text{秒}$  となる。一方、1年は、 $1 \text{年} = 365.25(\text{日}) \times 24(\text{時間}) \times 60(\text{分}) \times 60 \text{秒} = 31557600 \text{秒} = 3.15576 \times 10^7 \text{秒}$  であるから、完成するまでの年数は、

$$4 \times 10^9 \div 3.15576 \times 10^7 = \text{約 } 126.75(\text{年})$$

$$1 \text{年を } 365 \text{日とすると、} 126.84(\text{年})$$

(5) 【ヒント】 マイコンピュータのアイコンを右クリックして表示される「プロパティ」を選択すると「システムのプロパティ」が示され、利用しているPCのスペックを表示することができる。

(6) 調査時期により、最高速PCは異なるスペックのものとなる。

2010年夏の時点で市販PCの中で最高速のもの代表的なスペックは以下の通りである。

CPU: インテル Core i7-980X プロセッサ エクストリーム・エディション、クロック周波数 3.33 GHz(最大 3.6 GHz)、

L2 キャッシュ 1.5MB、L3 キャッシュ 12MB

メモリ容量: 6GB~24GB

HDD 容量: 500GB~2TB

大容量記憶装置: ブルーレイ記録 / DVD スーパーマルチドライブ

(7) 以下のように各ステップが並列処理される。したがって5つの命令が全て処理されるまでの時間は9ステップに要する時間となり、合計90ナノ秒となる。

命令 フェッチ	デコード	データ フェッチ	実行	ライト バック			
	命令 フェッチ	デコード	データ フェッチ	実行	ライト バック		
		命令 フェッチ	デコード	データ フェッチ	実行	ライト バック	
			命令 フェッチ	デコード	データ フェッチ	実行	ライト バック
				命令 フェッチ	デコード	データ フェッチ	ライト バック

(8) 1秒あたり処理する命令数が最も多いものが、プログラム処理時間が最も短いことになる。

1秒あたり処理する命令数は、クロック周波数×IPC で求めることができる。ア～エのCPUについての値は以下の通りである。

ア)  $0.9 \text{ GHz} \times 0.35 = 0.315$

イ)  $1.3 \text{ GHz} \times 0.25 = 0.325$

ウ)  $1.5 \text{ GHz} \times 0.22 = 0.33$

エ)  $2.0 \text{ GHz} \times 0.16 = 0.32$

この結果から、(ウ)のCPUが、プログラム処理時間が最も短いことがわかる。

(9) キャッシュメモリアクセスが90%で、メモリアクセスが10%となる。したがって平均呼び出し時間は以下の通りとなる。

$$50(\text{ナノ秒}) \times 0.9 + 450(\text{ナノ秒}) \times 0.1 = 90(\text{ナノ秒})$$

(10) ・格納しているデータを保持(リフレッシュ)する処理が不要なメモリであり、データを早く読み出せるので、キャッシュメモリとしてよく用いられる。

⇒ SRAM

・紫外線で全内容を消して書き直せるメモリである。

⇒ PROM

・主記憶装置に広く使われており、記憶内容を保つておくためにデータの再書き込みを常に行う必要がある。集積度が

高く、記憶容量あたりのコストが安い。

⇒ DRAM

・バックアップ電源が不要で、電氣的に全部または一部分を消して内容を書き直せるメモリである。

⇒ EEPROM(EPROM)

・製造時にプログラムやデータが書き込まれる。マイクロプログラムやゲームプログラムの供給に用いられる。

⇒ マスク ROM

(11)

	磁気ディスク	磁気テープ
容量	1 台あたり数 10GB~2TB と大容量である。複数の磁気ディスクを接続して論理的に 1 台と認識させることも可能である。コンピュータシステムがリアルタイムに利用するデータを供給することが可能である。単位容量あたりの価格は磁気テープに比べて高価。	1 巻あたり数 100MB~80GB である。用いるデータの大きさに応じて必要な容量の磁気テープを選択することが可能。単位容量あたりの価格は磁気ディスクに比べて安価。
アクセス方法	ランダムアクセス	シーケンシャルアクセス
アクセス速度	高速	低速
利用方法	プログラム、データの保管	データのバックアップ

(12) ハードディスク容量は以下の計算で求められる。

ディスク表面数×ディスク面上のセクタ総数×1セクタあたり容量

ここで

ディスク表面数 …… 2

ディスク面上のセクタ総数 ……  $2000 \times 600 + 4000 \times 450 = 12 \times 10^5 + 18 \times 10^5 = 3 \times 10^6$

1セクタあたり容量 …… 512B

よって、この HDD の容量は、以下の値になる。

$2 \times 3 \times 10^6 \times 512B = 3 \times 10^6 \times 1KB = 2.86GB$ (約 3GB)

【ヒント】1KB=1024B、1MB=1024KB、1GB=1024MB の関係を用いています。

(13) 1 回転に要する時間は 1 分間に 7200 回転であるから、

$60 \div 7200(\text{秒}) = 0.00833(\text{秒}) = 8.33(\text{ミリ秒})$

となる。ディスクの平均回転待ち時間は 1 回転の時間の 1/2 であるから 4.16 ミリ秒となる。

したがって HDD の平均アクセス時間は、平均シークタイムと平均回転待ち時間との合計であり、以下の通りとなる。

$8.5 + 4.16 = 12.7(\text{ミリ秒})$

(14) ・図形や写真などを入力する装置。OCR ソフトと連携して OCR の機能も実現する。

⇒ (イメージ)スキャナ

・タイムカードなどに刻印された文字を読み取る装置。数値やアルファベットなど手書きの文字も読み取ることが可能。

⇒ OCR

・PC で用いられる代表的な文字入力装置。キーを押すことによって入力文字の信号が伝えられる。

⇒ キーボード

・ボールの回転を 2 次元の移動距離に変換し、座標入力装置として用いるもの。

⇒ マウス

・POS システムで用いられ、商品コードなどを読み取る装置。

⇒ バーコードリーダー

・写真撮影されたフィルムのデータを画像情報として読み取る装置。

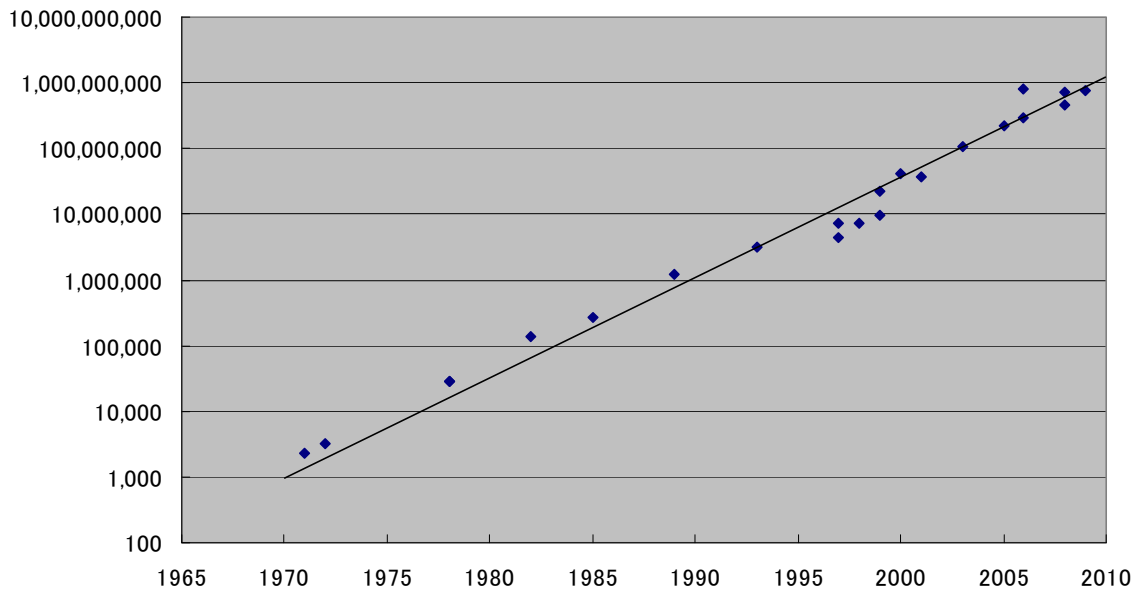
⇒ フィルムスキャナ

・特定様式の内紙上のマークを塗りつぶし、そのマークを読み取りデータ化する装置。

⇒ OMR

(15) 巻末資料にある CPU の発表年とトランジスタ数をグラフにしたのが次のグラフである。プロットされた点をつなぎ合わせると、ムーアの法則に従ってトランジスタ数が増加していることを確認することができる。なお、グラフ中に示されている直線は 24 ヶ月で倍増するとした場合の結果をあらわしたものである。

## トランジスタ数



【ヒント】ムーアの法則は「集積回路におけるトランジスタ数は18~24ヶ月で倍増する」という経験則である。このCPUのトランジスタ数を見た場合は18ヶ月より24ヶ月の方が実際の結果を良く表していると考えられる。

## 4章

- (1) OS の機能：ハードウェアを効率よく使用するための基本ソフトウェア。「4-3 OS の機能と構造」および表4-3などがヒントになる。PC の OS(WINDOWS)は、「ユーザが PC を使いやすくするため」というのがその根本的目的である。4-1-1 などの記述を参考にすること。
- (2) OS の誕生した経緯：「4-2 OS のはじまりとその進化」を熟読すること。高価なハードウェアを少しでも有効に利用しようと模索した苦勞の結果が OS である。
- (3) OS の主な機能：4-3 の①から⑩までが回答そのもの。また PC では、(1)同様、「ユーザが PC を使いやすくするための諸機能」があげられる。
- (4) 現在の代表的な OS: 表4-3 が回答そのもの。これに加えて PC のための OS、WINDOWS がある。
- (5) PC 用の OS：4-5 の記述がヒント。この章を充分熟読した上で、PC アーキテクチャを公開した IBM、IBM のベンダーとしてスタートしたマイクロソフト、WINDOWS の模範となったマッキントッシュ、これらの関係を読み取ることにより、その発展の経緯が明らかになる。
- (6) 日本の PC 開発のハードウェアの観点からの考察については、p.38「(5) 日本でのパーソナルコンピュータの歴史」参照。ソフトウェアの観点からの考察としては、次の4点にまとめることができよう。
  - ①当時の日本の PC 市場が未熟で WINDOWS のようなヒューマンインターフェイス・システムを受け入れる土壌に育っていなかったこと。
  - ②日本語の場合、多種文字言語である日本語の入力そのものが、PC を利用する際の最初のかつ大きなハードルになっていたため、これを解決したワープロソフトがそれだけで市場に受け入れられたこと。
  - ③当時のハードウェアの能力では、画面の品質、レスポンスタイムなどの面で WINDOWS の機能を十分に発揮できず、市場に受け入れてもらえなかったこと。
  - ④ ①に関連することであるが、市場が未熟であったため PC の利用者がごく一部の職業的ユーザに限定され、プログラム至上主義的な考え方が支配的であったこと。

## 5章

(1) ウォーターフォールモデルでは、要件定義、外部設計、内部設計と順に作業手順を踏んでいくため、要件定義や外部設計など最初のプロセスの成果が、後続するプロセスの品質を決めてしまうことになる。したがって、これら最初のプロセスを実施する際には、後続するプロセスで起こりうる作業や問題などを、実施に先立って予測しておく必要がある。このような予測は、経験豊富な SE にしかできない。これが、要件定義や外部設計など最初のプロセスを難しくしている要因である。

(2) 要件定義の段階では、システム化する業務システムそのものについて十分に熟知しておかなければならない。業務システムを形成する要因として例を挙げると、その企業の内部業務ルール、外部との取引ルールなどがある。さらにはその企業の業務風土などがシステム設計に影響をおよぼす可能性がある。また、設計・構築が終了してシステムが日常業務で使用されるときに、内部業務ルール、外部との取引ルール、業務風土などがシステム運用にどんな影響を与えるかについても考慮しておく必要がある。

(3) この場合のスコープマネジメントとは、情報システムを開発し導入するときに、

- ① 本来この情報システムを導入する目的は何か
- ② 一連の業務フローの中でどの部分をシステム化するのか
- ③ その期待される効果は何か

などを明確にすることである。さらにそれらを文書化して、関連する部門に対して事前に了承を得ておかなければならない。さもなければ、ユーザ部門の思惑が開発側と異なっていることがシステムを導入したあとになってわかった時、導入されたシステムがほとんど有効に使われないという事態になりうるからである。

(4) ウ

この問いに関しては、SQL の文法の詳細を知っておかなくてはならないが、本書では、次のような基本的な事項が理解できていれば充分である。

- ・ SQL がデータベース言語の代表的なものであること。
- ・ SQL では、決められた文法に従って記述された文を実行することにより、検索、挿入、編集、削除などのデータ操作が行われること。
- ・ エンドユーザにとって最も頻繁に使用するのが、データ検索に用いる SELECT 文であること。
- ・ 検索の基本は単純なパターンマッチングであり、データのもつ「意味」による検索はできないこと。

(5) 最初に、このデータレコードで、繰り返し項目になりそうなものを列挙する。ここでは、「宿泊日」以下移動交通手段までの項目になる。このうち、宿泊目的についてはさらにその内部で繰り返し項目になっているので、やや複雑である。まず、この宿泊目的については「どれか一つに限る」として考えると考えやすいだろう。

また、同姓同名の顧客の可能性もあるので、データベースではこの表にはない顧客 ID を新たに追加しておくのがよい。図 5 - 13 に対応した表ができるだろう。データ項目は、次のようになる。

- ・ 顧客 ID
- ・ 顧客名
- ・ 年齢
- ・ 住所
- ・ 電話番号
- ・ 電子メールアドレス
- ・ 料理、食材などの希望、好み
- ・ 宿泊日
- ・ 料理の追加特注
- ・ 宿泊部屋
- ・ 宿泊人数
- ・ 宿泊数
- ・ 担当係
- ・ 宿泊目的(温泉、景観、料理、ゴルフ、テニス、釣り、ドライブ、その他)のいずれか一つ
- ・ 移動交通手段(電車・バスなど、自家用車)

となる。何日も宿泊したお客さんについては、宿泊日ごとにこのレコードができることになる。

次に部分従属した項目を別の表にして、複数の表に分解する。例えば、

- ・ 顧客 ID
- ・ 顧客名
- ・ 年齢
- ・ 住所
- ・ 電話番号

- ・電子メールアドレス

の表と、

- ・顧客 ID
- ・宿泊日
- ・担当係
- ・料理、食材などの希望、好み
- ・料理の追加特注
- ・宿泊部屋
- ・宿泊人数
- ・宿泊目的(温泉、景観、料理、ゴルフ、テニス、釣り、ドライブ、その他)のいずれか一つ
- ・移動交通手段(電車・バスなど、自家用車)

の表が考えられる。

2 番目の表のうち、宿泊目的と移動交通手段については、次のような目的コードと交通手段コードを設定して、これを別表にするとわかりやすくなる。すなわち、表

- ・宿泊目的コード
- ・宿泊目的

と、表

- ・交通手段コード
- ・交通手段

などである。

## 6章

### (1) ア

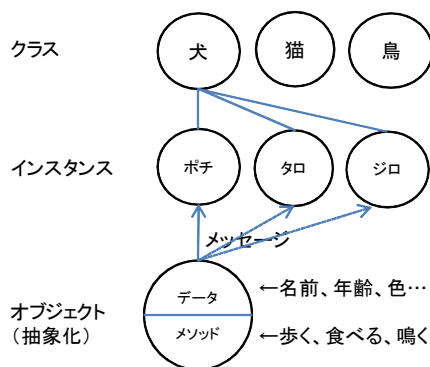
CはUNIXの構築に使用された言語であり、すべてのプログラムを再帰的(自分自身を呼び出すプログラム)に記述するわけでない。Fortranはプリプロセッサ(コンパイルの前に行う前処理のプログラム)をもつが、強力的ではない。Lispはリスト構造になっているが、データ型の宣言を厳格に行うのはPascalである。

### (2) ア

ソースプログラム → コンパイル → オブジェクトプログラム → リンク(リンカ) → ロードモジュール(ローダ) → 実行(本書 p.126 参照)

### (3) イ

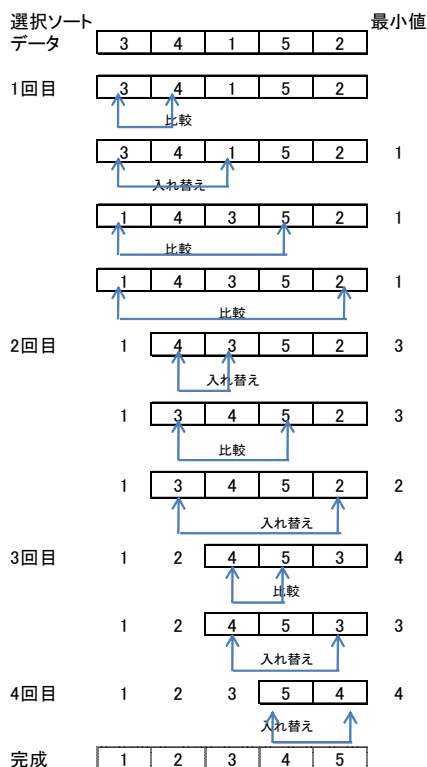
ある共通した性質をもつ「オブジェクト」を一つに集め、定義したものを「クラス」という。このクラスに属するオブジェクトのことを「インスタンス」という。共通性を認識できるように、オブジェクト間の不要なデータや属性を取り除いて、できあがったデータとメソッドの組み合わせを「抽象化」という。並列処理において複数のタスクに分割していく過程を「クラスタ化」という。



### (4) エ

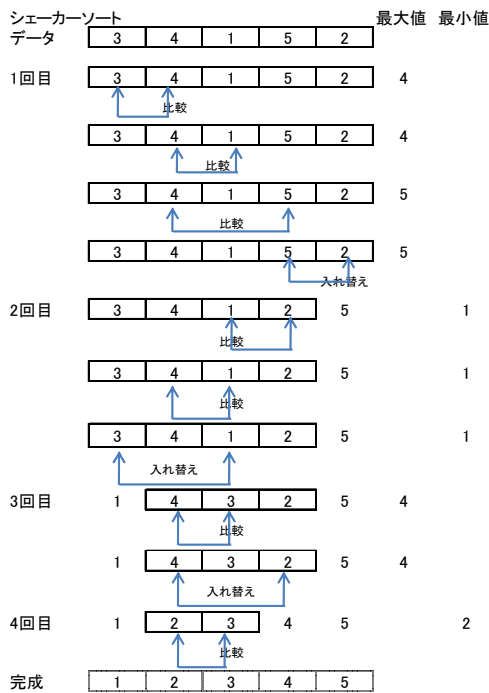
アはシェルソートのこと、イはバブルソートのこと、ウはクイックソートである(本書 p.131 参照)。

### (5)



本書 p.131 参照。

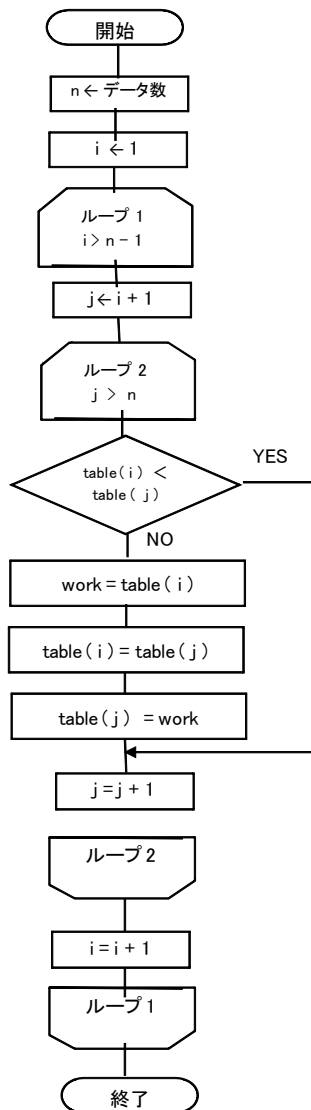
(6)



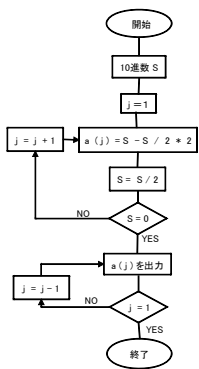
本書 p.131 参照。

(7)  $n$  個の数値(table)を選択ソート(基本選択法)するアルゴリズムを次に示す。

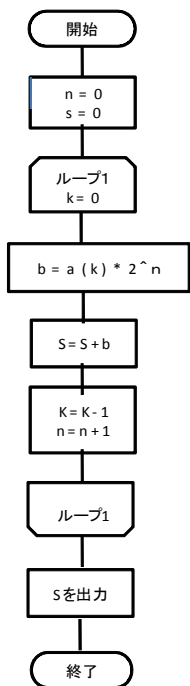
整列処理



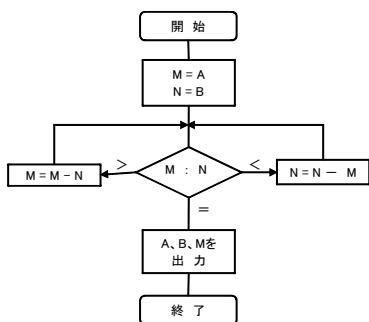
(8) 10進数(S)を2進数(a)に変換するアルゴリズムを次に示す。



(9) kケタの2進数(a)を10進数(S)に変換するアルゴリズムを次に示す。



(10) 2数AとBの最大公約数Mを探し出すアルゴリズムを次に示す。





## 7章

### (1) ウ

「サーバ」は「クライアント」からの要求に応じて処理するコンピュータであり、サーバとクライアントのOS(Operating System)は必ずしも同一種類でなくてもよく、1つのサーバで複数の機能を担うこともできる。

### (2) イ

「CSMA/CD」(Carrier Sense Multiple Access with Collision Detection)は伝送路上にデータがないかどうかを検知した上で送信する方式、同時に送信されると衝突がおこる。「トークリング」はケーブルを用いたLAN(Local Area Network)、「FDDI」(Fiber Distributed Data Inter face)は光ファイバを用いたLANで、いずれもトークンパッシング(送信権をえて伝送する)方式をとっている。「ATM」(Asynchronous Transfer Mode)は1本の回線を複数の論理回線に分割して通信する方法。

### (3) ウ

「ゲートウェイ」はネットワーク上で媒体やプロトコルが異なるデータを相互に変換して送信を可能にする機器。「ルータ」はネットワーク層のアドレスを見て、どの経路を通して転送すべきかを判断する機器。「ブリッジ」は転送先のアドレスを見て中継すべきかどうかを判断する機器でデータリンク層にあたる。

### (4) ア

「ADSL」(Asymmetric Digital Subscriber Line)は既存の電話回線を使って高速のデータ伝送を行い、イ、ウは「ISDN」(Integrated Services Digital Network)、エは「光ファイバ」通信である。

### (5) ウ

TCP/IP はデータを分割してパケットを作り、そのパケットにアドレスを付与してネットワークに送り出すしくみ。「TCP」(Transmission Control Protocol)はOSI参照モデル(Open System Interconnection)のトランスポート層、「IP」(Internet Protocol)はネットワーク層にあたる。

### (6) ウ

「FTP」はインターネット上でファイルを転送するプロトコル(File Transfer Protocol)、「HTML」はWebページを制作するプログラム言語(Hyper Text Markup Language)、「SMTP」は電子メールを送信するためのプロトコル(Simple Mail Transfer Protocol)である。

### (7) ア

公開鍵暗号方式は、インターネットで公開されている受信者の「公開鍵」で平文を暗号化して送信、受信者は暗号文を「秘密鍵」で平文に復号する方式である。

### (8) エ

コンピュータシステムの利用者に付与される情報は「ユーザID」(識別名)であり、「パスワード」は本人確認にもちいる暗証番号である。

### (9) ア

ウイルスに感染したディスクは、「ワクチン(コンピュータウイルス対策ソフト)」でウイルスを駆除するか、ディスクを論理フォーマットすればよい。

### (10) イ

電子メールを暗号化(SSL: Secure Socket Layer)して送信すれば、電子メールの内容の漏えいを防止することができる。