

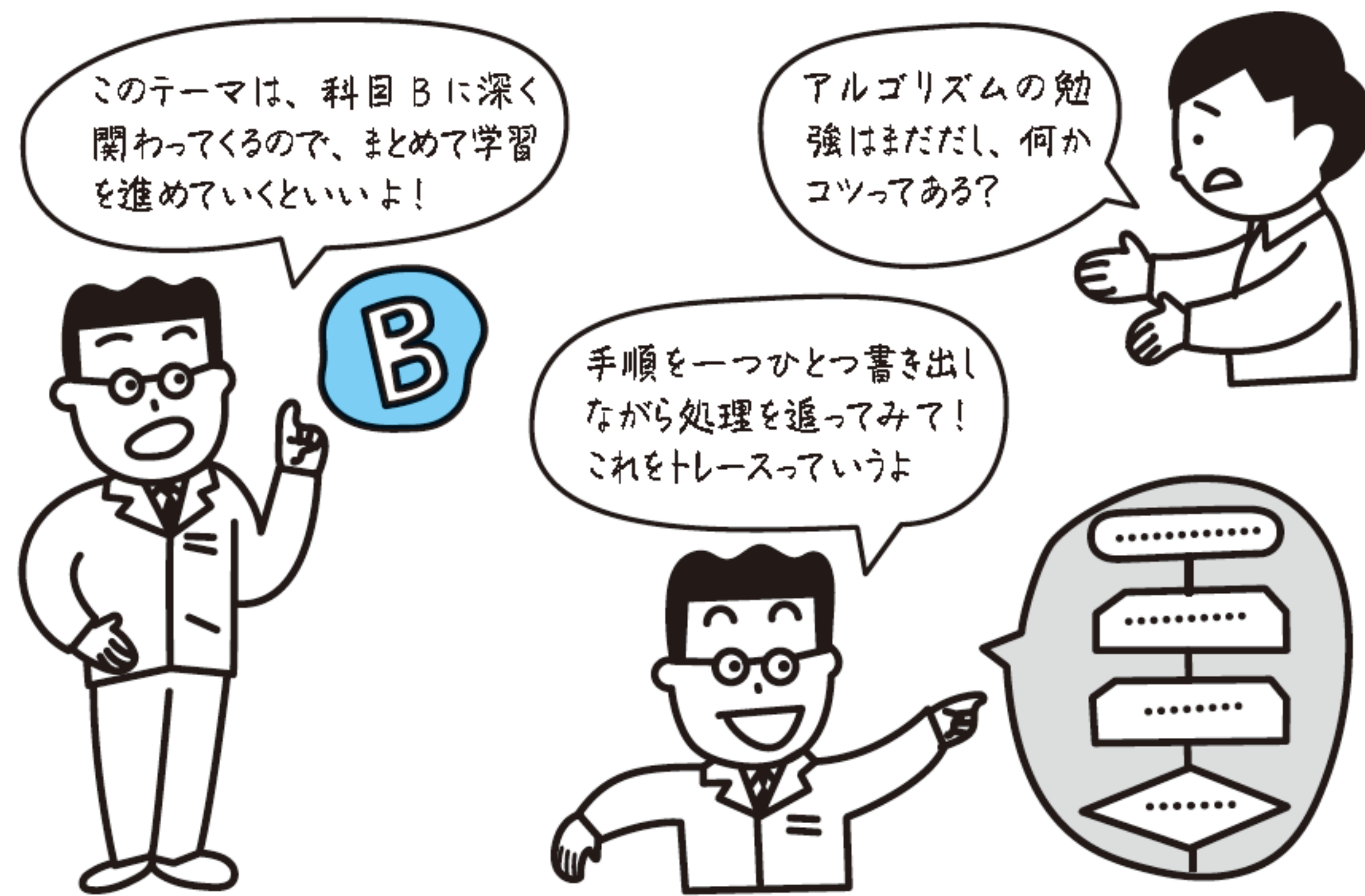
# 第2章の学習ガイダンス

## テクノロジ系

# データ構造とアルゴリズム

基本情報技術者試験では、科目Bでアルゴリズム（擬似言語プログラム）が重点的に出題されますが、科目Aでもアルゴリズム問題は出題されており、非常に重要視されていることがわかります。学習対策としては、流れ図の問題は必ず出ると考え、トレースできるようにしておきましょう。

また、キューやスタックなど、各データ構造を操作する問題もよく出ています。



1	データ構造の種類と特徴	44ページ
2	木構造と木の巡回	48ページ
3	流れ図のトレースと計算量の考え方	54ページ
4	データの整列アルゴリズム	58ページ
5	データの探索アルゴリズム	64ページ
6	再帰関数の呼出し	70ページ

## プログラミング未経験者なら、まず科目Bの学習から

基本情報は、科目Aと科目Bの両方が合格基準を超えないと合格できません。科目Bは、アルゴリズムとプログラミング分野の擬似言語の問題16問と情報セキュリティ分野の文章問題4問です。

プログラミングの経験がない場合は、科目Bの擬似言語の学習に十分時間をかけてください。そうすれば、科目Aのアルゴリズムの問題はスムーズに理解でき、解けるようになります。実務経験者の場合は、実務ではあまり使わない流れ図の見方などに目を通して、1~2問トレースしておけばよいでしょう。

### ●代表的なアルゴリズムの特徴を知っておこう

本書では、実務経験者や学習経験者が復習できる程度に、データ整列やデータ探索などの代表的なアルゴリズムの特徴を示しました。

全く聞いたことがないという人は、ネットで「バブルソート」などを検索してみてください。視覚的にわかりやすく解説しているサイトがたくさん見つかります。

## どのようなデータ構造か、特徴を理解しよう

データ構造の種類として、シラバスには次の5つが掲載されています。

- ① 配列
- ② リスト（線形リスト、単方向リスト、双方向リスト）
- ③ スタックとキュー（FIFO、LIFO、プッシュ、ポップ）
- ④ 木構造（2分木、完全2分木、2分探索木など）

まずは、各データ構造の特徴を理解することから始めましょう。

### ●スタックとキューの問題は図を描いて

スタックやキューを操作する問題がよく出ます。スタックやキューは構造が単純なので、図を描いて慎重に操作すれば誰でも容易に正解できる問題です。定規を使っていないに描く必要はありません。試験本番で時間配分に困らないよう、学習時に手早く簡単に図を描く練習をして慣れておきましょう。

### ●線形リストや2分木は科目Bでも出題される

平成5年度からの新試験を実施するにあたって、最初に公開された科目Bのサンプル問題には、線形リストや2分木のプログラムがありました。これらのデータ構造は科目Aの知識問題として学習するだけでなく、科目Bのプログラム問題として学習したほうが効果的です。



# 組合せと確率

基本情報技術者試験の出題範囲を示すシラバスの基礎理論の最初に書かれているのが数学です。なかでも確率はこれまでに何回も出題されていて、今後も出題される可能性が高いテーマといえます。なお数学の範囲は、過去に出題されていない項目も多いので、高校数学をすべて復習する必要はありませんよ。



## 組合せ

### ひらがなカードの組合せ

「あ」から「お」までの5枚のカードから2枚を引いたとき、引く順番には関係なく、2枚のカードの組合せは何通りでしょうか。



2枚ずつ書き出すと、(あ, い)、(あ, う)、(あ, え)、(あ, お)、(い, う)、(い, え)、(い, お)、(う, え)、(う, お)、(え, お)の10通りです。

中学で習った組合せ(Combination)の公式 ${}_n C_r$ が浮かんだ人も多いでしょう。

### 🕒 時短で覚えるなら、コレ!

#### 組合せの数

$n$ 個の中から $r$ 個を取り出す組合せの数は、次の式で表せる

$${}_n C_r = \frac{n!}{(n-r)! r!}$$

上記の公式を使うと、次のように10通りになります。

$${}_5 C_2 = \frac{5!}{(5-2)! 2!} = \frac{5 \times 4 \times 3 \times 2 \times 1}{(3 \times 2 \times 1) (2 \times 1)} = \frac{5 \times 4}{2 \times 1} = 10 \text{通り}$$

“!”で示される $n$ の階乗は、 $n$ から1までを掛け合わせたもの。3の階乗なら、 $3 \times 2 \times 1$ で、6になるよ。



## 場合の数と確率

### 確率といえば、まずはコインやサイコロ

まずは、次の例を解いてみましょう。

- ① コインを投げて、表が出る確率はいくらですか?
- ② サイコロを投げて、4の目が出る確率はいくらですか?

#### (考え方)

##### ① コインは2面あるから……



コインは表と裏の2面あり、表は1面なので確率は次のようになります。

$$\frac{1 \text{面}}{2 \text{面}} = \frac{1}{2}$$

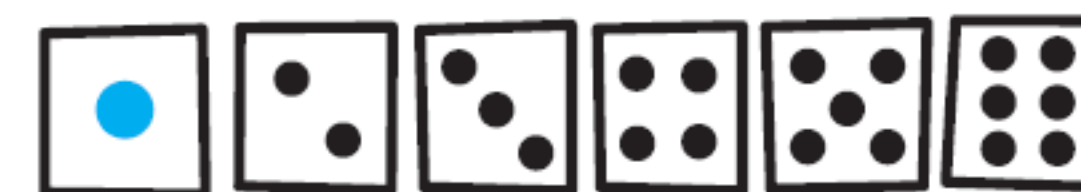
もしコインが4枚あれば、 $2 \times 2 \times 2 \times 2 = 16$ 通りの状態があります。2面あるコインは、1桁を0と1で表す2進数の考え方に通じるものです。

$$\bullet = 0 \quad \circ = 1$$

一般的なコンピュータでは、2進数で演算を行いますから、2進数4桁でも次のように16種類の並びを作ることができます。

0000、0001、0010、0011、0100、0101、0110、0111、  
1000、1001、1010、1011、1100、1101、1110、1111

0000を10進数の0とするので、2進数4桁では10進数の0から15までの16種類の値を表せることになります。



##### ② サイコロは6面あるから……

1から6までの6面中、4の目は1面なので確率は次のようになります。

$$\frac{1 \text{面}}{6 \text{面}} = \frac{1}{6}$$

コインやサイコロ投げのように繰り返すことができる操作や実験などを**試行**と呼び、試行した結果を**事象**といいます。例えば、サイコロを投げるのが試行で、4の目が出たというのが事象です。

事象の数を**場合の数**と呼びます。サイコロ全体には、1から6の目までの6つの事象があり、全体の場合の数は6です。4の目が出る場合の数は1なので、4の目が出る確率は、全体の場合の数6で割った $1/6$ になります。



解説

**明らかに異なるものを除外し、実際に並べ替えて解く**

この種の問題でクイックソート(次ページ参照)の具体例が求められる可能性は低いでしょう。したがって、残りの候補は3つです。実際に(4, 1, 3, 2)を並べ替えてみれば正解がわかります。

選択ソート	挿入ソート	バブルソート
4, 1, 3, 2	4, 1, 3, 2	4, 1, 3, 2
1, 4, 3, 2	1, 4, 3, 2	1, 4, 3, 2
1, 2, 4, 3	1, 3, 4, 2	1, 3, 4, 2
1, 2, 3, 4	1, 2, 3, 4	1, 3, 2, 4
		1, 3, 2, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4
		1, 2, 3, 4

注) 小さなものを選択

解答 ウ



**解いて覚える頻出用語 高度な整列アルゴリズム**

次の説明文と関連の深い用語を選べ。

- ある一定間隔おきに取り出した要素から成る部分列をそれぞれ整列し、更に間隔を詰めて同様の操作を行い、間隔が1になるまでこれを繰り返す。
- データ列の半分の容量の作業領域を用意し、データ列の分割、整列、併合を、繰り返す。
- 中間的な基準値を決めて、それよりも大きな値を集めた区分と小さな値を集めた区分に要素を振り分ける。
- 未整列の部分を順序木に構成し、そこから最大値または最小値を取り出して既整列の部分に移す。これらの操作を繰り返して、未整列部分を縮めていく。

- ア ヒープソート
- イ クイックソート
- ウ マージソート
- エ シェルソート

← これらの高度なソートは、科目Aでは並べ替えの特徴や計算量が問われる。プログラムが長くなりやすいため、小問形式の科目Bでも、出題しにくいはず。ただし、大まかなアルゴリズムは知っておくとよい。

解答 (1) エ (2) ウ (3) イ (4) ア

**確認のための実践問題**

問 クイックソートの処理方法を説明したものはどれか。

- ア 既に整列済みのデータ列の正しい位置に、データを追加する操作を繰り返していく方法である。
- イ データ中の最小値を求め、次にそれを除いた部分の中から最小値を求める。この操作を繰り返していく方法である。
- ウ 適当な基準値を選び、それより小さな値のグループと大きな値のグループにデータを分割する。同様にして、グループの中で基準値を選び、それぞれのグループを分割する。この操作を繰り返していく方法である。
- エ 隣り合ったデータの比較と入替えを繰り返すことによって、小さな値のデータを次第に端の方に移していく方法である。

**問の解説 …… 基準値というキーワードで解く**

ウに「基準値」とあるので、クイックソートと判断できます。その他の選択肢も考えてみてください。アは挿入ソート、イは選択ソート、エはバブルソート(交換ソート)です。

**クイックソート**は、**基準値(軸)との大小関係でデータを分類し、2つに分割することを再帰的に繰り返します**。基準値の決め方はいろいろありますが、ここでは並びの中央のデータを用いた例を示します。

1) 基準値を決める	8 6 4 7 3 1 5 2
分類と分割	2 6 4 5 3 1 7 8
2) 基準値を決める	2 6 4 5 3 1 7 8
分類と分割	2 1 3 4 5 6 7 8
3) 基準値を決める	2 1 3 4 5 6 7 8
分類と分割	1 2 3 4 5 6 7 8
4) 基準値を決める	1 2 3 4 5 6 7 8
分類と分割	1 2 3 4 5 6 7 8
整列終了	1 2 3 4 5 6 7 8

注) 確定したものを黒丸で示した。

解答 ウ



# 日程を短縮する手法

## 並行に作業したら早く終わるのではないかな?

ファストトラックは、本来は順々に実行する計画になっている作業を、なんとか工夫して並行して行うことで、期間を短縮するという技法です。言葉ではわかりにくいので、問題を使って具体的に見ていきましょう。

皆でやれば短縮できるって考え方だね!



## ファストトラック技法

ファストトラック技法を用いてスケジュールの短縮を行う。

当初の計画は図1のとおりである。作業Eを作業E1、E2、E3に分けて、図2のように計画を変更すると、スケジュールは全体で何日短縮できるか。

図1

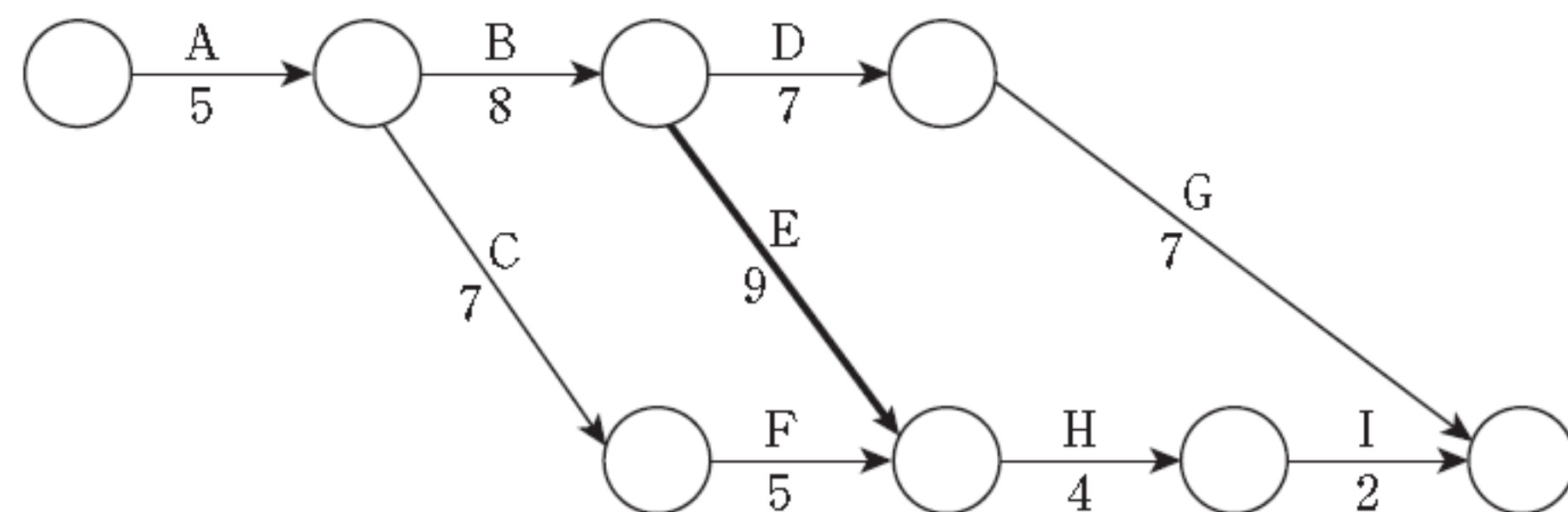
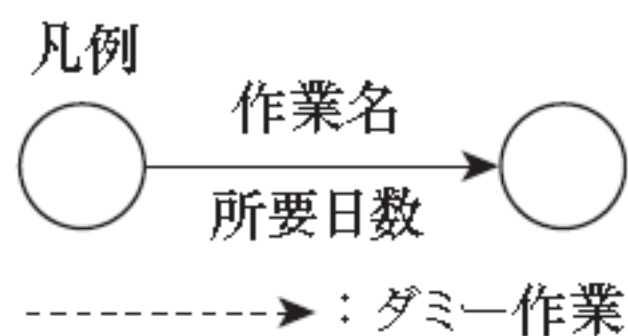
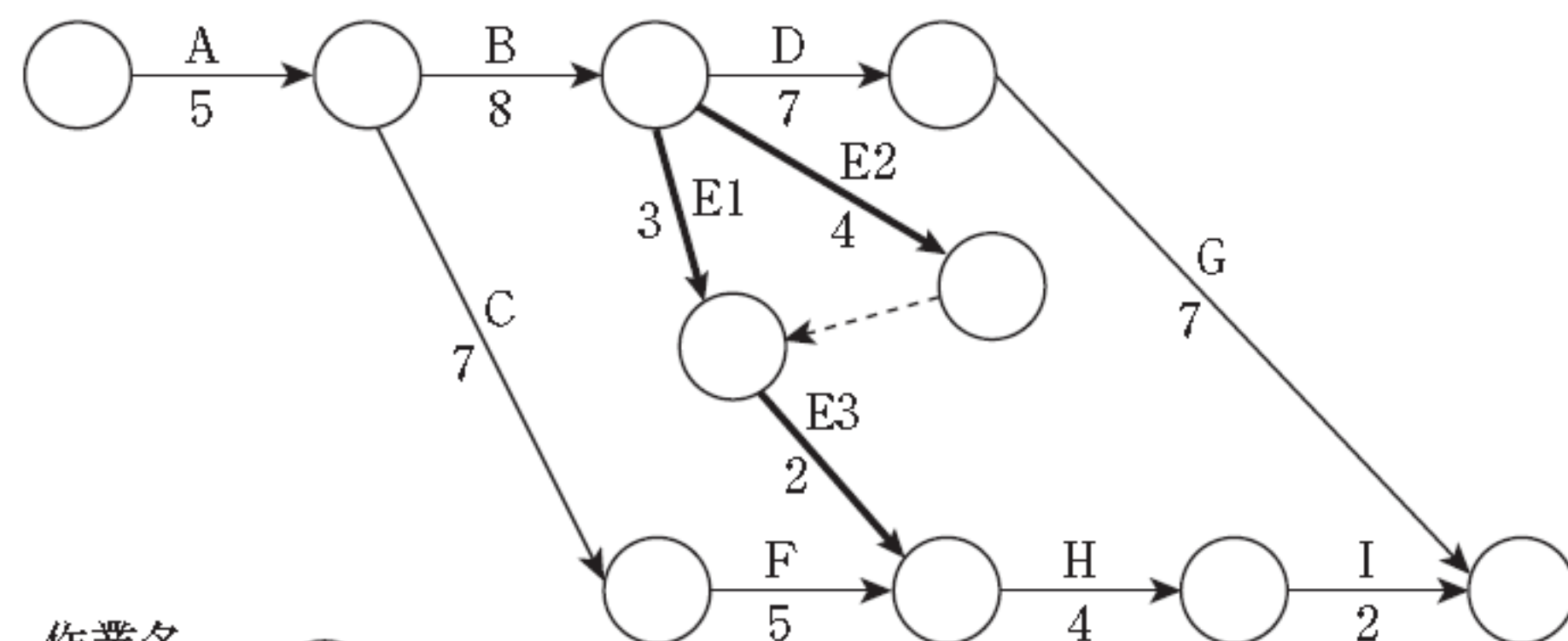


図2



ア 1          イ 2          ウ 3          エ 4

### 用語メモ

**ファストトラック (fast tracking)**

直列に並んでいる作業を、可能なものは並行して行うことで、期間(時間)を短縮する技法。

⤵ 並行作業に失敗するリスクあり

**クラッシング (Crashing)**

費用と期間(時間)のトレードオフを分析し、期間の短縮を少ない追加費用で実現し、費用の最適化を図る技法。

⤵ 開発要員の追加など



## 作業Eを3つに分解して並列に作業することで、全体の期間を短縮

まず、図1に作業日数を足しながら書き込んで、当初の日数を計算していきましょう。A→B→D→Gが27日、A→B→E→H→Iが28日です。A→C→F→H→Iの作業Fまでは17日で終わりますが、作業Eの終了を待って作業Hを始めるので、28日かかります。3つの経路の中で、日程の余裕がなく、最も期間のかかるA→B→E→H→Iがクリティカルパスです。

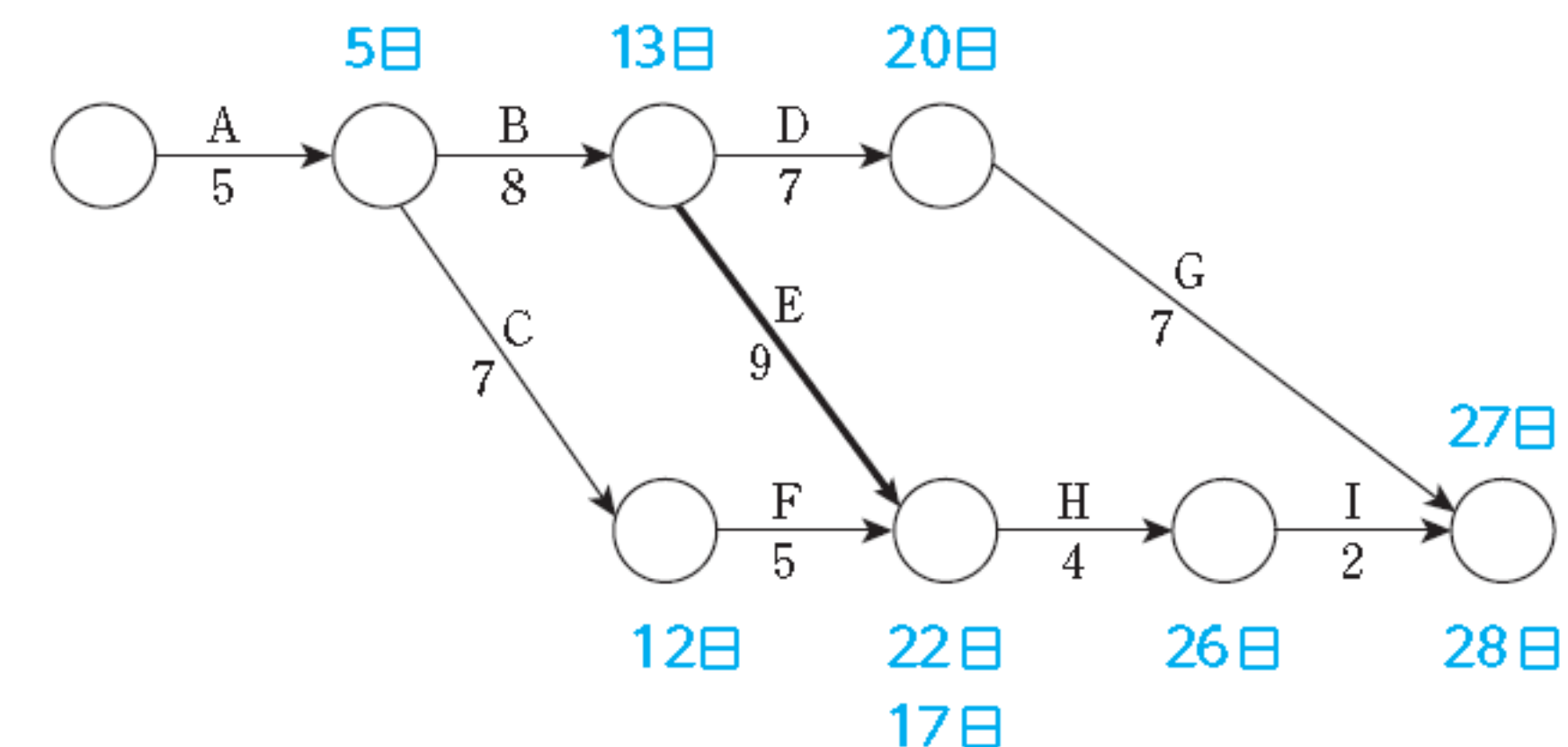
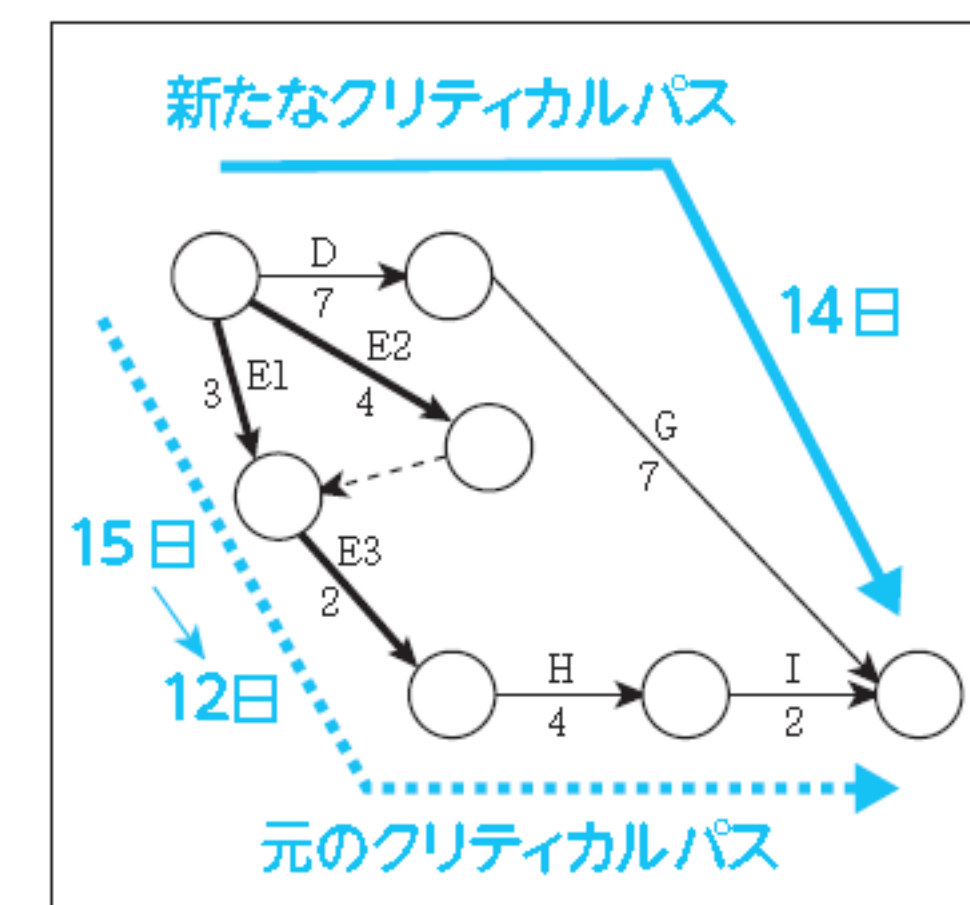


図2では、作業Eを3つに分解して並列に作業することで、全体の期間を短縮します。図1のEは9日で行っていましたが、図2はE1とE2を並列に行うことで、6日に短縮できます。図1でクリティカルパス上にあった作業Eが2日以上短縮されるため、27日かかるA→B→D→Gが新たなクリティカルパスになります。つまり、作業Eを短縮しても、スケジュール全体では1日しか短縮されません。

3日短縮と早とちりしないよう注意



時短で覚えるなら、コレ!

### クリティカルパス その2

作業の短縮によって、クリティカルパスが変わることがある



# 章末問題

出典：基本情報技術者試験  
科目A試験サンプル問題

解説動画  
p.6

## 出題例 キャッシュメモリ

目標解答時間 2分

問 A～Dを、主記憶の実効アクセス時間が短い順に並べたものはどれか。

	キャッシュメモリ			主記憶
	有無	アクセス時間 (ナノ秒)	ヒット率 (%)	アクセス時間 (ナノ秒)
A	なし	—	—	15
B	なし	—	—	30
C	あり	20	60	70
D	あり	10	90	80

ア A、B、C、D

イ A、D、B、C

ウ C、D、A、B

エ D、C、A、B

### ●解説

#### 順序を求める問題は、大小関係を考えれば計算しなくてもOK!

AとBはキャッシュメモリがないので、主記憶のアクセス時間から、 $A < B$ 。またAとCを比べると、CのキャッシュでさえAの主記憶より遅いので、 $A < C$ 。

さらにCとDを比べると、Dは90%を10ナノ秒でアクセスできるので、主記憶がCより少し遅くても、 $D < C$ です。 $A < B$ 、 $A < C$ 、 $D < C$ に該当するのは、イだけです。このように理屈で考えれば、時間の節約になります。

なお、計算でも求めてみましょう。93ページの式を使って、CとDの実効アクセス時間を計算します。

$$\begin{aligned} C &: 20 \text{ナノ秒} \times 0.6 + 70 \text{ナノ秒} \times (1 - 0.6) \\ &= 12 \text{ナノ秒} + 28 \text{ナノ秒} = \mathbf{40 \text{ナノ秒}} \end{aligned}$$

$$\begin{aligned} D &: 10 \text{ナノ秒} \times 0.9 + 80 \text{ナノ秒} \times (1 - 0.9) \\ &= 9 \text{ナノ秒} + 8 \text{ナノ秒} = \mathbf{17 \text{ナノ秒}} \end{aligned}$$

したがって、次のようになります。

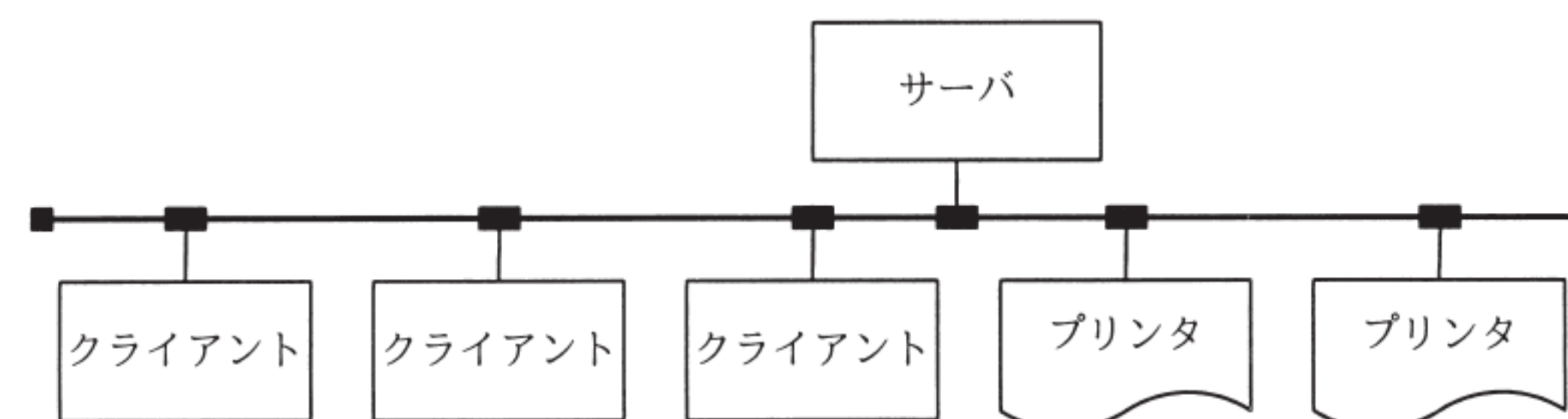
$$\mathbf{A (15 \text{ナノ秒}) < D (17 \text{ナノ秒}) < B (30 \text{ナノ秒}) < C (40 \text{ナノ秒})}$$

【解答】 イ

## 出題例 システムの稼働率

目標解答時間 2分

問 図のように、1台のサーバ、3台のクライアントおよび2台のプリンターがLANで接続されている。このシステムはクライアントからの指示に基づいて、サーバにあるデータをプリンターに出力する。各装置の稼働率が表のとおりであるとき、このシステムの稼働率を表す計算式はどれか。ここで、クライアントは3台のうちどれか1台が稼働していればよく、プリンターは2台のうちどちらかが稼働していればよい。



装置	稼働率
サーバ	$a$
クライアント	$b$
プリンタ	$c$
LAN	1

ア  $ab^3c^2$

イ  $a(1-b^3)(1-c^2)$

ウ  $a(1-b)^3(1-c)^2$

エ  $a(1-(1-b)^3)(1-(1-c)^2)$

### ●解説

#### クライアントとプリンターは並列システムとして考える

・「クライアントは3台のうちどれか1台が稼働していればよく」  $1 - (1-b)^3$

・「プリンターは2台のうちどちらかが稼働していればよい」  $1 - (1-c)^2$

サーバは1つなので、直列システムと考えると、式は次のようになります。

$$\mathbf{a \times (1 - (1-b)^3) \times (1 - (1-c)^2)}$$

【解答】 エ