

3.1 アルゴリズムとプログラミング(1)

データ構造

問1 【解答イ】

ア：配列は、同じ形式のデータをまとめて取り扱うデータ構造である。

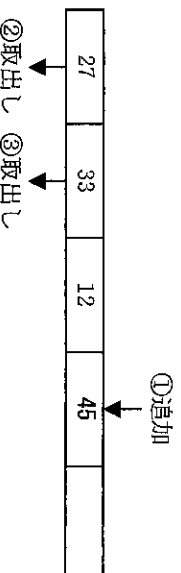
イ：配列は、一つひとつのデータ（要素）を番号（添字または指標）で区別する。そのため、添字によってデータを任意の順序で読み出すことができる。（正解）

ウ：リストは、データを先頭要素（先頭データ）から順番にたどるデータ構造である。そのため、先頭データ以外を直接参照することはできない。

エ：リストは、データの追加や削除はポインタの値を変更するだけでよい。そのため、既存のデータを移動する必要はない。

問2 【解答ウ】

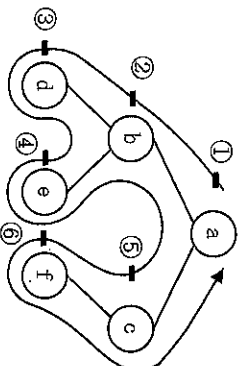
キューは、先に記録したデータを先に読み出す、先入れ先出し（FIFO: First-In First-Out）方式のデータ構造である。問題文の追加・取出しの流れを、図に示す。



したがって、2番目に取り出されるのは「33」である。

問3 【解答イ】

木構造は、データを階層構造（親子関係）で表すデータ構造である。階層の上位から下位に向かって、節（ノード）をたどること、データを取り出すことができる。問題の取出し方法に従って、節、左部分木、右部分木の順にデータを取り出していくと「a, b, d, e, c, f」となる。

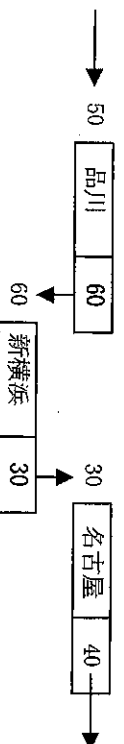


問4 【解答ア】

問題のリストを図で表すと、次のようになる。



このリストの“品川”と“名古屋”の間に“新横浜”を挿入すると、次のようになる。



したがって、「新横浜」のポイントが30とし，“品川”のポイントが60とする。」

問5 【解答ウ】

ステップにA→B→C→Dの順に入力し，解答群の順に出力する操作を考えていくと，次のようになる。なお，[]内が出力されるデータを表している。

ア：A①→②[A]→B①→C①→D①→②[D]→②[C] …Bは取り出せない
 イ：A①→B①→②[B]→C①→D①→②[D]→②[C] …Aは取り出せない
 ウ：A①→B①→C①→②[C]→②[B]→D①→②[D]→②[A] (正解)
 エ：A①→B①→C①→D①→②[D]→②[C]→②[B] …Aは取り出せない

3.1 アルゴリズムとプログラミング(2)

アルゴリズム

問1 【解答イ】

アルゴリズムは，「コンピュータに，ある特定の目的を達成させるための処理手順」のことである。アルゴリズム(処理手順)は，流れ図(フローチャート)などの図式を用いて視覚的に分かりやすく表現する。

ア：言語プログラミングに関する説明である。

ウ：プログラミング言語に関する説明である。

エ：CAD (Computer Aided Design；コンピュータ支援設計) に関する説明である。

問2 【解答ウ】

処理をトレースしていくと，次のようになる。

- ① 処理 “1 → x” : 変数 x に 1 を代入する。(x=1)
- ② 処理 “3 → x” : 変数 x に 3 を代入する。(x=3)
- ③ 条件 “x > y” : 条件を判定する。
→ x=1, y=3なので条件は成立しない。(Noに分岐する)
- ④ 処理 “y - x → z” : y - x = 3 - 1 = 2 を変数 z に代入する。
- ⑤ 処理 “z を出力” : 変数 z の値「2」を出力する。

問3 【解答イ】

ア：繰返し構造は，条件によって処理を繰り返す構造である。

イ：繰返し構造で使用される終了条件は，成立するまで処理を繰り返す条件である。一方，成立している間，処理を繰り返す条件は継続条件という。(正解)

ウ：前判定型繰返し構造の場合，最初の判定で繰返しを抜けて，繰返し中の処理を一度も実行しない場合がある。

エ：選択構造は，条件によって処理を分岐させる構造である。

問 4 【解答イ】

A=1, B=2を初期値として, 実際にトレースしてみる。

ア:

| | TMP | A | B |
|---------|-----|---|---|
| 初期値 | | 1 | 2 |
| TMP ← A | 1 | 1 | 2 |
| A ← B | 1 | 2 | 2 |
| B ← A | 1 | 2 | 2 |

イ:

| | TMP | A | B |
|---------|-----|---|---|
| 初期値 | | 1 | 2 |
| TMP ← A | 1 | 1 | 2 |
| A ← B | 1 | 2 | 2 |
| B ← TMP | 1 | 2 | 1 |

ウ:

| | TMP | A | B |
|---------|-----|---|---|
| 初期値 | | 1 | 2 |
| TMP ← B | 2 | 1 | 2 |
| A ← B | 2 | 2 | 2 |
| B ← A | 2 | 2 | 2 |

エ:

| | TMP | A | B |
|---------|-----|---|---|
| 初期値 | | 1 | 2 |
| TMP ← B | 2 | 1 | 2 |
| A ← B | 2 | 2 | 2 |
| B ← TMP | 2 | 2 | 2 |

したがって, データが正しく入れ替わる (A=2, B=1となる) のは「イ」の手順である。

問 5 【解答ウ】

〔手続き〕に従って, 処理を行うと次のようになる。

| | x | y |
|-----|-------|-------|
| (1) | 2 | 3 |
| (2) | 2 | 3-1=2 |
| (3) | 2+2=4 | 2 |
| (4) | 4 | 2 |
| (2) | 4 | 2-1=1 |
| (3) | 4+1=5 | 1 |
| (4) | 5 | 1 |

(y=1) → 終了

したがって, 処理が終了したとき, xの値は「5」となる。

3.1 アルゴリズムとプログラミング(3)

合計アルゴリズム

問 1 【解答エ】

フローチャート (流れ図) の作成手順は, 次のとおりである。

(1) 問題分析

入力データ, 出力データ, データ構造などを確認・分析する。

(2) アルゴリズムの作成

問題分析の結果をもとに, 必要な処理手順を考える。

(3) 流れ図の作成

(2)で作成したアルゴリズムを, 流れ図 (フローチャート) で表す。

したがって, 「問題分析 → アルゴリズムの作成 → 流れ図の作成」となる。なお, アルゴリズムの作成と流れ図の作成を並行して行うこともあるが, 問題分析は必ず最初に行われる。

問 2 【解答ウ】

この流れ図では最後に変数xを出力しているので, 1~Nの整数の合計は変数xに求める。そのためには, 変数xに加算している変数yを, 1, 2, 3, ..., Nと変えていくことを考える。したがって, 空欄aの条件がYes (真) の間, 処理を繰り返すので, 「y ≤ N」と初期値1から1ずつ増加する変数yがN以下である間, 加算処理「x + y → x」を繰り返すようにする。