

EXAMPRESS®

情報処理技術者試験学習書

特別抜粋版



[対応試験: FE]

出るところだけ!

基本情報技術者

科目B

橋本祐史

SE
SHOEISHA

予想 + 過去問題集

売上1位の参考書に完全対応した

待望の

科目B問題集

- ✓ 全107問(擬似言語74問, 情報セキュリティ33問)を解説。
- ✓ 良質なオリジナル予想問題で, 充実した問題演習ができる。

ご案内

本資料は、2024年9月25日発売の書籍『情報処理教科書 出るところだけ！基本情報技術者 [科目 B] 予想+過去問題集』から一部を抜粋し、特別に編集したものです。

全書は『情報処理教科書 出るところだけ！基本情報技術者 [科目 B] 予想+過去問題集』でご覧いただけます。購入および、詳しい書籍情報は以下のリンクからどうぞ。

Amazon販売ページ



<https://www.amazon.co.jp/dp/479818781X>

書籍情報

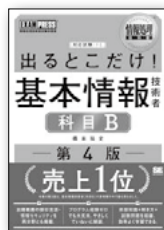


<https://www.shoeisha.co.jp/book/detail/9784798187815>

はじめに

本書は『情報処理教科書 出るとこだけ！ 基本情報技術者【科目B】
第4版』に完全対応した問題集です。この姉妹書（**テキスト**）の読者が、受験を前に、実践的なトレーニングをするための**問題集**です。

科目Bで合格基準点600点（1,000満点）を獲得できる率が**50%弱**に低迷するなか、「**テキスト**で基礎力を、**問題集**で実践力を」というコンセプトのもと執筆しました。



● たくさん問題を解きたい人へ

「トレースを練習するための問題が足りない」という声をよく耳にします。同書（**テキスト**）に収録された問題をすべてやり終えたため、練習する問題が不足しているようです。擬似言語で問われるのは**知識**の有無でなく、**技能**です。技能は繰り返し行う訓練により、時間をかけて身につけるものです。そのために必要なのは、数多くの問題を解くことです。

「1問5分では、解答時間が足りない」という声もよく聞きます。試験では膨大な量のトレースを短時間でこなす必要があります。速く解くために必要なものと同じく、数多くの問題を解くことです。

本書の**全107問**（予想問題3回分・公開問題3回分など。情報セキュリティを含む）により、充実した問題演習ができます。

● 良質な予想問題で合格への実践力が身につく

試験対策の**王道**は「基本的な問題で誤答しない」ことです。つまり難問対策には時間や労力をかけずに、正解できる問題で正解を積み重ねることで、合格をたぐり寄せます。

本書では「難しすぎず、やさしすぎない」レベルの、つまり試験で問われるちょうどよい難易度のオリジナル予想問題や改題を作成し、収録しました。

本書が、読者のみなさんの合格の一助になれば、幸いです。

2024年8月
橋本 祐史

予想問題 1



基礎力を確認するための問題を掲載しました。初めはうまく解けない問題もあるかもしれませんが、試験形式で取り組むのではなく、「次回解けるように」を目標に、初めから解説を読み、解説をもとにみずからトレースするとよいでしょう。

科目	科目B	出題数	20問 (全問必須解答)	合格 基準点	600点 (1,000点満点)	試験 時間	100分
----	-----	-----	-----------------	-----------	--------------------	----------	------

問1

②一次元配列

解説 p.062

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。ここで、配列の要素番号は1から始まる。

関数 func を呼び出すと、戻り値は である。

[プログラム]

- ```

1: ○整数型の配列: func()
2: 整数型の配列: x ← {3, 1, 2, 4}
3: 整数型の配列: y ← {2, 1, 4, 3}
4: x[y[2]] ← x[y[4]]
5: x[y[x[1]]] ← y[x[y[2]]]
6: return x

```

## 解答群

- ア {1, 1, 2, 4}    イ {2, 1, 2, 4}    ウ {2, 1, 4, 3}    エ {3, 1, 2, 4}

次の記述中の  と  に入れる正しい答えの組合せを、解答群の中から選べ。

関数 func を func(175, 77) として呼び出すと、`/** α **/` の行を  回実行し、`/** β **/` の行を  回実行する。

[プログラム]

```

1: ○整数型: func(整数型: num1, 整数型: num2)
2: 整数型: x ← num1
3: 整数型: y ← num2
4: while (x ≠ y)
5: if (x > y)
6: x ← x - y /** α **/
7: else
8: y ← y - x /** β **/
9: endif
10: endwhile
11: return x

```

解答群

|   | a | b |
|---|---|---|
| ア | 2 | 2 |
| イ | 2 | 3 |
| ウ | 2 | 4 |
| エ | 3 | 2 |
| オ | 3 | 3 |
| カ | 3 | 4 |

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。

手続 test を呼び出すと、出力は  となる。

[プログラム]

11: ○整数型: func1(整数型: x, 整数型: y)

12: return x + y

21: ○整数型: func2(整数型: x, 整数型: y)

22: return x ÷ y

31: ○整数型: func3(整数型: x, 整数型: y)

32: return x × y

41: ○整数型: func4(整数型: x, 整数型: y)

42: return x - y

51: ○test()

52: 整数型: ret

53: ret ← func1(func2(4, 2), func3(func4(5, 3), func1(1, 2)))

54: ret を出力する

解答群

ア 1

イ 2

ウ 3

エ 4

オ 5

カ 6

キ 7

ク 8

ケ 9

コ 10

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。

手続programAを呼び出したとき、出力は  の順となる。

[プログラム]

1: 大域: 文字型: a ← "A"

11: ○programA()

12: a ← "B"

13: a を出力する

14: 文字型: a ← "A"

15: programB(a)

16: programC()

17: a を出力する

21: ○programB(文字型: b)

22: a を出力する

23: b を出力する

24: 文字型: a ← b

25: a を出力する

31: ○programC()

32: a を出力する

33: a ← "C"

34: a を出力する

解答群

ア "A", "A", "B", "A", "A", "A", "A"

ウ "A", "A", "B", "A", "A", "C", "A"

オ "B", "B", "A", "A", "B", "C", "A"

キ "B", "B", "B", "B", "B", "A", "B"

イ "A", "A", "B", "A", "A", "A", "B"

エ "A", "A", "B", "B", "A", "C", "B"

カ "B", "B", "A", "A", "B", "C", "C"

ク "B", "B", "B", "B", "B", "C", "B"



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。ここで、配列の要素番号は1から始まる。

手続 `isosceles` を呼び出して処理が終了した直後の `data` の内容は  である。

[プログラム]

```
1: 大域: 整数型の二次元配列: data ← {3行3列の 0の値}
2: ○ isosceles()
3: 整数型: i, j, v ← 1
4: for (i を data の行数 から 1 まで 1 ずつ減らす)
5: for (j を i から data の列数 まで 1 ずつ増やす)
6: data[i, j] ← v
7: v ← v + 1
8: endfor
9: endfor
```

解答群

- ア {{4, 5, 6}, {0, 2, 3}, {0, 0, 1}}
- イ {{4, 5, 6}, {3, 2, 0}, {0, 0, 1}}
- ウ {{4, 0, 0}, {5, 2, 0}, {6, 3, 1}}
- エ {{4, 3, 0}, {5, 2, 0}, {6, 0, 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。

関数  $r$  を  $r(4)$  として呼び出すと、戻り値は  である。

[プログラム]

```
1: ○整数型: r(整数型: x)
2: if (x < 2)
3: return 1
4: else
5: return r(x - 1) + r(x - 2)
6: endif
```

解答群

ア 3

イ 4

ウ 5

エ 6

オ 7

カ 8

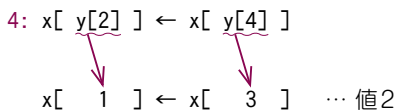
キ 9

ク 12

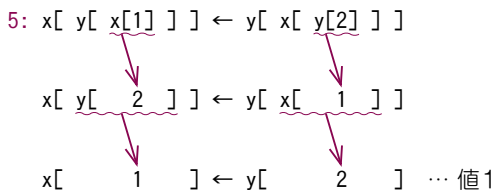
プログラム中に  がなく、トレースの結果が正解になる問題です。

|   | トレース表                            | 条件式 | x                                                                                                                               | y                                                                                                                               |   |   |   |   |   |   |   |   |
|---|----------------------------------|-----|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|---|---|---|---|
| A | 1: ○整数型の配列: func()               |     |                                                                                                                                 |                                                                                                                                 |   |   |   |   |   |   |   |   |
| B | 2: 整数型の配列: x ← {3, 1, 2, 4}      |     | <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>3</td><td>1</td><td>2</td><td>4</td></tr> </table> | 1                                                                                                                               | 2 | 3 | 4 | 3 | 1 | 2 | 4 |   |
| 1 | 2                                | 3   | 4                                                                                                                               |                                                                                                                                 |   |   |   |   |   |   |   |   |
| 3 | 1                                | 2   | 4                                                                                                                               |                                                                                                                                 |   |   |   |   |   |   |   |   |
| C | 3: 整数型の配列: y ← {2, 1, 4, 3}      |     |                                                                                                                                 | <table border="1"> <tr><td>1</td><td>2</td><td>3</td><td>4</td></tr> <tr><td>2</td><td>1</td><td>4</td><td>3</td></tr> </table> | 1 | 2 | 3 | 4 | 2 | 1 | 4 | 3 |
| 1 | 2                                | 3   | 4                                                                                                                               |                                                                                                                                 |   |   |   |   |   |   |   |   |
| 2 | 1                                | 4   | 3                                                                                                                               |                                                                                                                                 |   |   |   |   |   |   |   |   |
| D | 4: x[y[2]] ← x[y[4]]             |     | <table border="1"> <tr><td>2</td><td></td><td></td><td></td></tr> </table>                                                      | 2                                                                                                                               |   |   |   |   |   |   |   |   |
| 2 |                                  |     |                                                                                                                                 |                                                                                                                                 |   |   |   |   |   |   |   |   |
| E | 5: x[y[x[1]]] ← y[x[y[2]]]       |     | <table border="1"> <tr><td>1</td><td></td><td></td><td></td></tr> </table>                                                      | 1                                                                                                                               |   |   |   |   |   |   |   |   |
| 1 |                                  |     |                                                                                                                                 |                                                                                                                                 |   |   |   |   |   |   |   |   |
| F | 6: return x ← 戻り値は {1, 1, 2, 4}。 |     | <table border="1"> <tr><td>1</td><td>1</td><td>2</td><td>4</td></tr> </table>                                                   | 1                                                                                                                               | 1 | 2 | 4 |   |   |   |   |   |
| 1 | 1                                | 2   | 4                                                                                                                               |                                                                                                                                 |   |   |   |   |   |   |   |   |

- D行で[]の最も内側から順に値を当てはめる。x[1]にx[3]の値2を格納する。



- E行で[]の最も内側から順に値を当てはめる。x[1]にy[2]の値1を格納する。なお、x[1]にはD行で値2が格納されている。



- F行でプログラムが終了したため、配列xのすべての要素に値を書き込む。【こう解く 配列のトレース】(⇒ テキスト p.100)

よって、正解はアです。

- Q1 「5: x[y[x[2]]] ← y[x[y[4]]]」に変更した場合の正しい答えを、解答群の中から選べ。
- Q2 「5: x[y[x[3]]] ← y[x[y[3]]]」に変更した場合の正しい答えを、解答群の中から選べ。

## 問2 カ

やさしい 問題 p.035

この問のポイントは、次のとおりです。

## ◆関係演算子

試験問題では、同じ役割の2種類の表記により表されますが、どちらも同じ意味です。なお、トレース表では記号(「≠」や「>」など)を用いています。すばやく記入するためです。[関係演算子] (⇒ **テキスト** p.030)

- while (x ≠ y)
  - while (x が y と等しくない)
- ↳ どちらも同じ意味。
- if (x > y)
  - if (x が y より大きい)
- ↳ どちらも同じ意味。

プログラム中に  がなく、トレースの結果が正解になる問題です。

|   | トレース表                                | 条件式        | num1 | num2 | x   | y  |
|---|--------------------------------------|------------|------|------|-----|----|
| A | 1: ○整数型: func( 整数型: num1, 整数型: num2) |            | 175  | 77   |     |    |
| B | 2: 整数型: x ← num1                     |            |      |      | 175 |    |
| C | 3: 整数型: y ← num2                     |            |      |      |     | 77 |
| D | 4: while (x ≠ y)                     | 175 ≠ 77 T |      |      |     |    |
| E | 5: if (x > y)                        | 175 > 77 T |      |      |     |    |
| F | 6: x ← x - y                         |            |      |      | 98  |    |
| G | 4: while (x ≠ y)                     | 98 ≠ 77 T  |      |      |     |    |
| H | 5: if (x > y)                        | 98 > 77 T  |      |      |     |    |

|   | トレース表                   | 条件式            | num1 | num2 | x  | y  |
|---|-------------------------|----------------|------|------|----|----|
| I | 6: $x \leftarrow x - y$ |                |      |      | 21 |    |
| J | 4: while ( $x \neq y$ ) | $21 \neq 77$ T |      |      |    |    |
| K | 5: if ( $x > y$ )       | $21 > 77$ F    |      |      |    |    |
| L | 8: $y \leftarrow y - x$ |                |      |      |    | 56 |
| M | 4: while ( $x \neq y$ ) | $21 \neq 56$ T |      |      |    |    |
| N | 5: if ( $x > y$ )       | $21 > 56$ F    |      |      |    |    |
| O | 8: $y \leftarrow y - x$ |                |      |      |    | 35 |
| P | 4: while ( $x \neq y$ ) | $21 \neq 35$ T |      |      |    |    |
| Q | 5: if ( $x > y$ )       | $21 > 35$ F    |      |      |    |    |
| R | 8: $y \leftarrow y - x$ |                |      |      |    | 14 |
| S | 4: while ( $x \neq y$ ) | $21 \neq 14$ T |      |      |    |    |
| T | 5: if ( $x > y$ )       | $21 > 14$ T    |      |      |    |    |
| U | 6: $x \leftarrow x - y$ |                |      |      | 7  |    |
| V | 4: while ( $x \neq y$ ) | $7 \neq 14$ T  |      |      |    |    |
| W | 5: if ( $x > y$ )       | $7 > 14$ F     |      |      |    |    |
| X | 8: $y \leftarrow y - x$ |                |      |      |    | 7  |
| Y | 4: while ( $x \neq y$ ) | $7 \neq 7$ F   |      |      |    |    |
| Z | 11: return x ●——戻り値は7。  |                |      |      |    |    |

このうち、[/\*\*\*  $\alpha$  \*\*\*/]がある「6:  $x \leftarrow x - y$ 」は、3回実行しています。また[/\*\*\*  $\beta$  \*\*\*/]がある「8:  $y \leftarrow y - x$ 」は、4回実行しています。よって、正解は**カ**です。

▶ 別の値でトレースしよう

正解 p.016

- Q1 関数 func を func(77, 63) として呼び出した場合の正しい答えを、解答群の中から選べ。
- Q2 関数 func を func(87, 54) として呼び出した場合の正しい答えを、解答群の中から選べ。



カッコ内のカッコ内（途中でカッコがあるため）

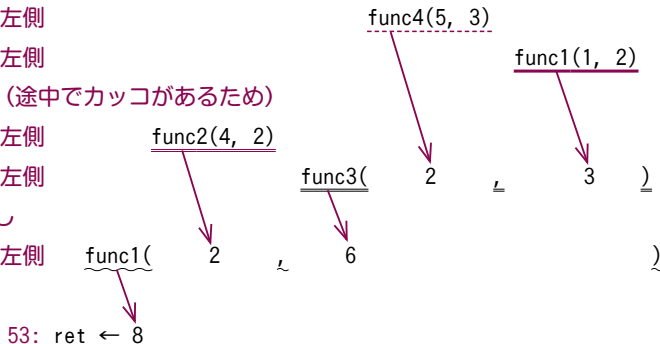
- ① 最も左側
- ② 次に左側

カッコ内（途中でカッコがあるため）

- ③ 最も左側
- ④ 次に左側

カッコなし

- ⑤ 最も左側



よって、正解はクです。

▶ 別の値でトレースしよう

正解 p.016

- Q 1 「53: ret ← func3(func4(4, 2), func3(func2(5, 3), func3(1, 2)))」に変更した場合の正しい答えを、解答群の中から選べ。
- Q 2 「53: ret ← func2(func2(12, 2), func4(func3(2, 3), func1(2, 1)))」に変更した場合の正しい答えを、解答群の中から選べ。

問 5

イ

やさしい 問題 p.038

この問のポイントは、次のとおりです。

◆ たいいき 大域変数

見分け方・有効範囲・トレースでの注意点は、次のとおりです。【大域変数】

(→ テキスト p.058)

- 見分け方：「大域:」付きで宣言された変数。プログラムの冒頭で宣言されることが多い。
- 有効範囲：どこからでも利用できる。
- トレース：プログラム実行前に最初に初期値が格納される。すべてのプログラム（関数・手順）が終了しても、大域変数の値は残り、そのまま再利用できる。

ここでは「1: 大域: 整数型: num ← 5」が該当します。

この問のポイントは、次のとおりです。

#### ◆ 局所変数と大域変数

局所変数と大域変数の違いをまとめると、次のとおりです。[局所変数と大域変数]

(→ テキスト p.059)

- **局所変数**は、プログラムが終了すると、局所変数の値はすべて消去され、再利用はできない。
- **大域変数**は、プログラム実行前に最初に初期値が格納される。すべてのプログラムが終了しても、大域変数の値は残り、そのまま利用できる。


#### ◆ 変数名の重複

局所変数・引数・大域変数の名前が重複する場合の考え方は、次のとおりです。

- 局所変数・引数・大域変数の名前が重複する場合、**局所変数・引数**の値を用いる。
- つまり、大域変数sと局所変数sが両方とも存在する状態で、「sを出力する」を実行すると、出力するのは局所変数sの値である。

なお、局所変数と引数は、名前が重複することはありません。

#### ◆ 別のプログラムのトレース

あるプログラム（関数・手続）から別のプログラムへのトレースの注意点は、次のとおりです。[ 別のプログラムのトレース] (→ テキスト p.059)

- 呼出し先のトレース表を右に**字下げ**して書く。呼出し元と呼出し先の関係を理解しやすくするために。
- **大域変数**の値は、すべての**トレース表**をまたいで使用する。
- 局所変数・引数・大域変数の名前が重複する場合、トレース表では大域変数の列名に「大域」を付ける。
- 例えば、大域変数sと局所変数sが両方とも存在するプログラムでは、トレース表では大域変数は「大域」を付け、「大域s」と記入する。



プログラム中に  がなく、トレースの結果が正解になる問題です。

11～17行が手続programA, 21～25行が手続programB, 31～34行が手続programCであり、3つのプログラムが記述されています。「1: 大域: 文字型: a ← "A"」で、大域変数はプログラム実行前に最初に初期値が格納されます。その後、手続programAを呼び出します。

|    | トレース表              | 条件式 | 大域a | a | 出力 |
|----|--------------------|-----|-----|---|----|
| AA | 1: 大域: 文字型 a ← "A" |     | A   |   |    |
| AB | 11: OprogramA()    |     |     |   |    |
| AC | 12: a ← "B"        |     | B   |   |    |
| AD | 13: a を出力する        |     |     |   | B  |
| AE | 14: 文字型: a ← "A"   |     |     | A |    |
| AF | 15: programB(a)    |     |     |   |    |

- AC行で大域変数aに値Bを格納する。この段階では局所変数aはprogramA中に存在しないため、大域変数aに格納する。
- AD行で出力するのはAC行で格納された大域変数aの値B。
- AE行で局所変数aを宣言するとともに、値Aを格納する。
- AF行で引数となるのはAE行で格納された局所変数aの値A。「局所変数・引数・大域変数の名前が重複する場合、局所変数・引数の値を用いる」ため。AF行でprogramB(a)として手続programBを呼び出す。


|    | トレース表                 | 条件式 | 大域a | b | a | 出力 |
|----|-----------------------|-----|-----|---|---|----|
| BA | 21: OprogramB(文字型: b) |     |     | A |   |    |
| BB | 22: a を出力する           |     |     |   |   | B  |
| BC | 23: b を出力する           |     |     |   |   | A  |
| BD | 24: 文字型: a ← b        |     |     |   | A |    |
| BE | 25: a を出力する           |     |     |   |   | A  |

- BB行で出力するのは、AC行で格納された大域変数aの値B。「大域変数の値は、すべてのトレース表をまたいで使用する」ため。【ごう解く 別のプログラムのトレース】(→ p.142)
- BC行で出力するのは、引数bの値A。
- BD行で局所変数aを宣言するとともに、引数bの値Aを格納する。
- BE行で出力するのは、AF行と同じ理由でBD行で格納された局所変数aの値A。programB()を終了し、programA()の続きを実行する。

|    | トレース表          | 条件式 | 大域a | a | 出力 |
|----|----------------|-----|-----|---|----|
| AG | 16: programC() |     |     |   |    |

- AG行で手続programCを呼び出す。

|    | トレース表           | 条件式 | 大域a | 出力 |
|----|-----------------|-----|-----|----|
| CA | 31: OprogramC() |     |     |    |
| CB | 32: aを出力する      |     |     | B  |
| CC | 33: a ← "C"     |     | C   |    |
| CD | 34: aを出力する      |     |     | C  |

- CB行で出力するのは、大域変数aの値B。「大域変数の値は、すべてのトレース表をまたいで使用する」ため。【 別のプログラムのトレース】(⇒ p.142)
- CC行で大域変数aに値Cを格納する。局所変数aはprogramC中に存在しないため、大域変数aに格納する。
- CD行で出力するのはCC行で格納された大域変数aの値C。programC()を終了し、programA()の続きを実行する。

|    |             |  |  |  |   |
|----|-------------|--|--|--|---|
| AH | 17: a を出力する |  |  |  | A |
|----|-------------|--|--|--|---|

- AH行で出力するのはAE行で格納された局所変数aの値A。「局所変数・引数・大域変数の名前が重複する場合、局所変数・引数の値を用いる」ため。programA()を終了する。

出力は“B”，“B”，“A”，“A”，“B”，“C”，“A”の順です。  
よって、正解は**オ**です。

▶ 別の値でトレースしよう

正解 p.016

- Q1 「14: 文字型: a ← "B"」に変更した場合の正しい答えを、解答群の中から選べ。

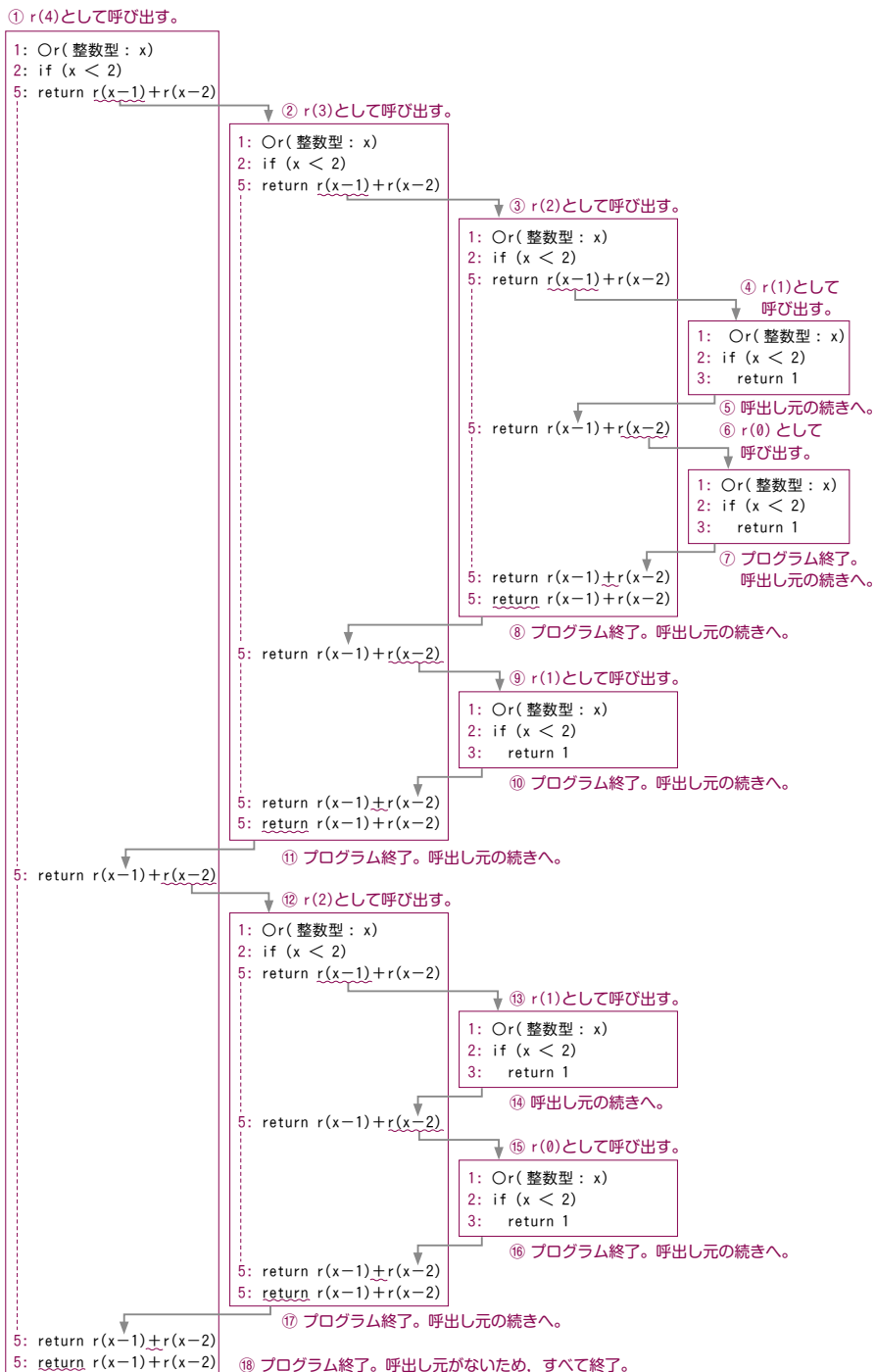
プログラム中に  がなく、トレースの結果が正解になる問題です。

|   | トレース表                                   | 条件式          | data                                                                                                                                                                                                                                                                                      | i | j | v |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
|---|-----------------------------------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|
| A | 1: 大域: 整数型の二次元配列:<br>data ← {3行3列の 0の値} |              | <table border="1" style="text-align: center; border-collapse: collapse;"> <tr><td></td><td>1</td><td>2</td><td>3</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>2</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>3</td><td>0</td><td>0</td><td>0</td></tr> </table> |   | 1 | 2 | 3 | 1 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 3 | 0 | 0 | 0 |  |  |  |
|   | 1                                       | 2            | 3                                                                                                                                                                                                                                                                                         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 1 | 0                                       | 0            | 0                                                                                                                                                                                                                                                                                         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 2 | 0                                       | 0            | 0                                                                                                                                                                                                                                                                                         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 3 | 0                                       | 0            | 0                                                                                                                                                                                                                                                                                         |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| B | 2: Oisosceles()                         |              |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| C | 3: 整数型: i, j, v ← 1                     |              |                                                                                                                                                                                                                                                                                           |   |   | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| D | 4: for(i を data の行数から1まで1ずつ減らす)         | $3 \geq 1$ T |                                                                                                                                                                                                                                                                                           | 3 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| E | 5: for(j を i から data の列数まで1ずつ増やす)       | $3 \leq 3$ T |                                                                                                                                                                                                                                                                                           |   | 3 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| F | 6: data[i, j] ← v                       |              | <table border="1" style="text-align: center; border-collapse: collapse;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>                                                                                | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 0            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 0            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 1            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| G | 7: v ← v + 1                            |              |                                                                                                                                                                                                                                                                                           |   |   | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| H | 5: for(j を i から data の列数まで1ずつ増やす)       | $4 \leq 3$ F |                                                                                                                                                                                                                                                                                           |   | 4 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| I | 4: for(i を data の行数から1まで1ずつ減らす)         | $2 \geq 1$ T |                                                                                                                                                                                                                                                                                           | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| J | 5: for(j を i から data の列数まで1ずつ増やす)       | $2 \leq 3$ T |                                                                                                                                                                                                                                                                                           |   | 2 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| K | 6: data[i, j] ← v                       |              | <table border="1" style="text-align: center; border-collapse: collapse;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>                                                                                | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 1 |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 0            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 2                                       | 0            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 1            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| L | 7: v ← v + 1                            |              |                                                                                                                                                                                                                                                                                           |   |   | 3 |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| M | 5: for(j を i から data の列数まで1ずつ増やす)       | $3 \leq 3$ T |                                                                                                                                                                                                                                                                                           |   | 3 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| N | 6: data[i, j] ← v                       |              | <table border="1" style="text-align: center; border-collapse: collapse;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>                                                                                | 0 | 0 | 0 | 0 | 2 | 3 | 0 | 0 | 1 |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 0            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 2                                       | 3            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |
| 0 | 0                                       | 1            |                                                                                                                                                                                                                                                                                           |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |  |  |

この時点で、該当する選択肢は1つに絞られます。

よって、正解は**ア**です。

再帰の構造図は、次のとおりです。このプログラムの [5: return r (x - 1) + r (x - 2)] は **【問題5-3】** (⇒ **テキスト** p.169) の6~8行を1行にまとめています。なお、図中の「    」はプログラム1行で複数の処理を実行する処理における実行部分を意味します。



トレース表は、次のとおりです。

- 問題文のとおり、最初に  $r(4)$  として関数  $r$  を呼び出す。

|    | トレース表                                               | 条件式       | x |
|----|-----------------------------------------------------|-----------|---|
| AA | 1: ○整数型: $r(\text{整数型}: x)$                         |           | 4 |
| AB | 2: $\text{if } (x < 2)$                             | $4 < 2$ F |   |
| AC | 5: $\text{return } \underline{r(x - 1)} + r(x - 2)$ |           |   |

- AC行で「 $\text{return } r(x - 1) + r(x - 2)$ 」のうちの「 $r(x - 1)$ 」を実行する。 $r(3)$  として関数  $r$  を呼び出す。

|    | トレース表                                               | 条件式       | x |
|----|-----------------------------------------------------|-----------|---|
| BA | 1: ○整数型: $r(\text{整数型}: x)$                         |           | 3 |
| BB | 2: $\text{if } (x < 2)$                             | $3 < 2$ F |   |
| BC | 5: $\text{return } \underline{r(x - 1)} + r(x - 2)$ |           |   |

- BC行で「 $\text{return } r(x - 1) + r(x - 2)$ 」のうちの「 $r(x - 1)$ 」を実行する。 $r(2)$  として関数  $r$  を呼び出す。

|    | トレース表                                               | 条件式       | x |
|----|-----------------------------------------------------|-----------|---|
| CA | 1: ○整数型: $r(\text{整数型}: x)$                         |           | 2 |
| CB | 2: $\text{if } (x < 2)$                             | $2 < 2$ F |   |
| CC | 5: $\text{return } \underline{r(x - 1)} + r(x - 2)$ |           |   |

- CC行で「 $\text{return } r(x - 1) + r(x - 2)$ 」のうちの「 $r(x - 1)$ 」を実行する。 $r(1)$  として関数  $r$  を呼び出す。

|    | トレース表                           | 条件式       | x |
|----|---------------------------------|-----------|---|
| DA | 1: ○整数型: $r(\text{整数型}: x)$     |           | 1 |
| DB | 2: $\text{if } (x < 2)$         | $1 < 2$ T |   |
| DC | 3: $\text{return } 1$ ●— 戻り値は1。 |           |   |

- DC行で  $\text{return}$  により、値 1 を  $r(1)$  の戻り値として返す。 $r(1)$  を終了し、 $r(2)$  の続きを実行する。

|    |                                                     |  |  |
|----|-----------------------------------------------------|--|--|
| CD | 5: $\text{return } r(x - 1) + \underline{r(x - 2)}$ |  |  |
|----|-----------------------------------------------------|--|--|

- CD行で「return r(x - 1) + r(x - 2)」のうちの「r(x - 2)」を実行する。r(2)におけるxの値は2のため、r(0)として関数rを呼び出す。

|    | トレース表                     | 条件式     | x |
|----|---------------------------|---------|---|
| EA | 1: ○整数型: r(整数型: x)        |         | 0 |
| EB | 2: if (x < 2)             | 0 < 2 T |   |
| EC | 3: return 1    ●—— 戻り値は1。 |         |   |

- EC行でreturnにより、値1をr(0)の戻り値として返す。r(0)を終了し、r(2)の続きを実行する。

|    | トレース表                                              | 条件式 | x |
|----|----------------------------------------------------|-----|---|
| CE | 5: return r(x - 1) + r(x - 2)                      |     |   |
| CF | 5: <u>return</u> r(x - 1) + r(x - 2)    ●—— 戻り値は2。 |     |   |

- CE行で「return r(x - 1) + r(x - 2)」のうちの「+」を実行する。つまり、r(1)の戻り値1 + r(0)の戻り値1 = 値2を求める。
- CF行でreturnにより、値2を戻り値として返す。r(2)を終了し、r(3)の続きを実行する。

|    | トレース表                                | 条件式 | x |
|----|--------------------------------------|-----|---|
| BD | 5: return r(x - 1) + <u>r(x - 2)</u> |     |   |

- BD行で「return r(x - 1) + r(x - 2)」のうちの「r(x - 2)」を実行する。r(3)におけるxの値は3のため、r(1)として関数rを呼び出す。

|    | トレース表                     | 条件式     | x |
|----|---------------------------|---------|---|
| FA | 1: ○整数型: r(整数型: x)        |         | 1 |
| FB | 2: if (x < 2)             | 1 < 2 T |   |
| FC | 3: return 1    ●—— 戻り値は1。 |         |   |

- FC行でreturnにより、値1をr(1)の戻り値として返す。r(1)を終了し、r(3)の続きを実行する。

|    |                                                    |  |  |
|----|----------------------------------------------------|--|--|
| BE | 5: return r(x - 1) + r(x - 2)                      |  |  |
| BF | 5: <u>return</u> r(x - 1) + r(x - 2)    ●—— 戻り値は3。 |  |  |

- BE行で「return r(x - 1) + r(x - 2)」のうちの「+」を実行する。つまり、r(2)の戻り値2 + r(1)の戻り値1 = 値3を求める。

- BF行でreturnにより、値3を戻り値として返す。r(3)を終了し、r(4)の続きを実行する。

|    | トレース表                                | 条件式 | x |
|----|--------------------------------------|-----|---|
| AD | 5: return r(x - 1) + <u>r(x - 2)</u> |     |   |

- AD行で「return r(x - 1) + r(x - 2)」のうちの「r(x - 2)」を実行する。r(4)におけるxの値は4のため、r(2)として関数rを呼び出す。

|    | トレース表                                | 条件式     | x |
|----|--------------------------------------|---------|---|
| GA | 1: ○整数型: r(整数型: x)                   |         | 2 |
| GB | 2: if (x < 2)                        | 2 < 2 F |   |
| GC | 5: return <u>r(x - 1)</u> + r(x - 2) |         |   |

- GC行で「return r(x - 1) + r(x - 2)」のうちの「r(x - 1)」を実行する。r(1)として関数rを呼び出す。

|    | トレース表                    | 条件式     | x |
|----|--------------------------|---------|---|
| HA | 1: ○整数型: r(整数型: x)       |         | 1 |
| HB | 2: if (x < 2)            | 1 < 2 T |   |
| HC | 3: return 1     ●—戻り値は1。 |         |   |

- HC行でreturnにより、値1をr(1)の戻り値として返す。r(1)を終了し、r(2)の続きを実行する。

|    |                                      |  |  |
|----|--------------------------------------|--|--|
| GD | 5: return r(x - 1) + <u>r(x - 2)</u> |  |  |
|----|--------------------------------------|--|--|

- GD行で「return r(x - 1) + r(x - 2)」のうちの「r(x - 2)」を実行する。r(2)におけるxの値は2のため、r(0)として関数rを呼び出す。

|    | トレース表                    | 条件式     | x |
|----|--------------------------|---------|---|
| IA | 1: ○整数型: r(整数型: x)       |         | 0 |
| IB | 2: if (x < 2)            | 0 < 2 T |   |
| IC | 3: return 1     ●—戻り値は1。 |         |   |

- IC行でreturnにより、値1をr(0)の戻り値として返す。r(0)を終了し、r(2)の続きを実行する。

|    | トレース表                                  | 条件式 | x        |
|----|----------------------------------------|-----|----------|
| GE | 5: return $r(x - 1) + r(x - 2)$        |     |          |
| GF | 5: <u>return</u> $r(x - 1) + r(x - 2)$ |     | ← 戻り値は2。 |

- GE行で「return  $r(x - 1) + r(x - 2)$ 」のうちの「+」を実行する。つまり、 $r(1)$ の戻り値1 +  $r(0)$ の戻り値1 = 値2を求める。
- GF行でreturnにより、値2を戻り値として返す。 $r(2)$ を終了し、 $r(3)$ の続きを実行する。

|    | トレース表                                  | 条件式 | x        |
|----|----------------------------------------|-----|----------|
| AE | 5: return $r(x - 1) + r(x - 2)$        |     |          |
| AF | 5: <u>return</u> $r(x - 1) + r(x - 2)$ |     | ← 戻り値は5。 |

- AE行で「return  $r(x - 1) + r(x - 2)$ 」のうちの「+」を実行する。つまり、 $r(4)$ の戻り値3 +  $r(2)$ の戻り値2 = 値5を求める。
- AF行でreturnにより、値5を戻り値として返す。 $r(4)$ を終了し、プログラムを終了する。

よって、正解はウです。

▶ 別の値でトレースしよう 正解 p.016

- Q 1 「5: return  $r((x + 1) \div 2) + r(x \div 2)$ 」に変更した場合の正しい答えを、解答群の中から選べ。
- Q 2 「5: return  $r((x + 2) \div 3) + r((x \div 3) + 1)$ 」に変更した場合の正しい答えを、解答群の中から選べ。

**問 9** カ やさしい 問題 p.195

この問のポイントは、次のとおりです。二分木については **【二分木】** (→ **テキスト** p.174) を参照してください。

**◆二分木のたどり方**

二分木の各節を一度だけたどるためのたどり方には、大きく分けて**深さ優先探索**と**幅優先探索**があります。さらに深さ優先探索には、次の3つのたどり方があります。