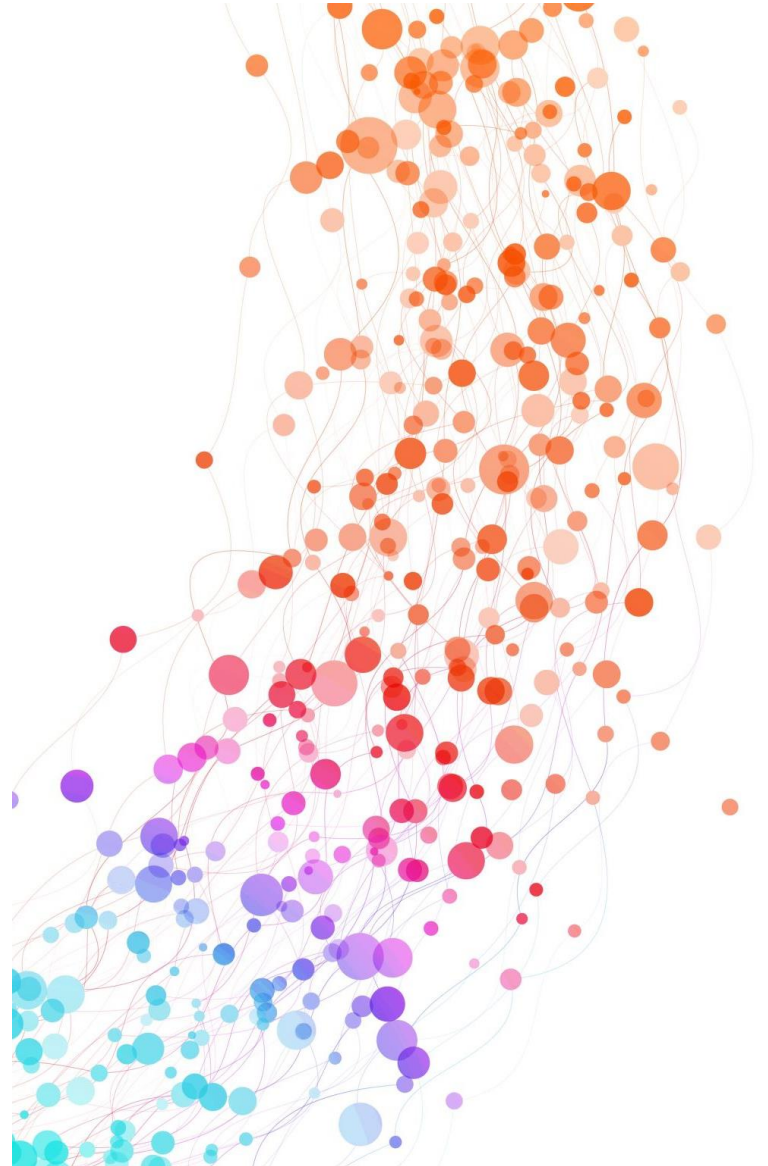


Benesse模試_ 配列解説

3年情報



まずAさんは配列を利用し、クラス名を **Kurasumei**、獲得点を **Tensu** として、表1のデータを格納することにした。また、全体のクラス数の値を変数 **kurasu_num** に格納した。このようにして図1のプログラムを書いたAさんは、想定したデータが正しく出力されるか試してみることにした。その結果、**Tensu[4]** は と予想どおりに表示された。

なお、すべての配列の添字は 0 から始まっているものとする。例えば、**Kurasumei[0]** の値は "1A" である。

```
(1) Kurasumei = ["1A", "1B", "1C", "1D", "2A", "2B", "2C",  
                "2D", "3A", "3B", "3C", "3D"]  
(2) Tensu = [42, 24, 13, 27, 11, 49, 65, 67, 54, 60, 65, 3]  
(3) kurasu_num = 
```

図1 データを配列と変数に格納するプログラム

・ の解答群

① 3 ② 4 ③ 11 ④ 12 ⑤ 13 ⑥ 27 ⑦ 49

●添字0から始まる

答え ア ②

まずAさんは配列を利用し、クラス名を `Kurasumei`、獲得点を `Tensu` として、表1のデータを格納することにした。また、全体のクラス数の値を変数 `kurasu_num` に格納した。このようにして図1のプログラムを書いたAさんは、想定したデータが正しく出力されるか試してみることにした。その結果、`Tensu[4]` は と予想どおりに表示された。

なお、すべての配列の添字は0から始まっているものとする。例えば、`Kurasumei[0]` の値は "1A" である。

```
(1) Kurasumei = ["1A", "1B", "1C", "1D", "2A", "2B", "2C",
                 "2D", "3A", "3B", "3C", "3D"]
(2) Tensu = [42, 24, 13, 27, 11, 49, 65, 67, 54, 60, 65, 3]
(3) kurasu_num = 
```

図1 データを配列と変数に格納するプログラム

・ の解答群

① 3 ② 4 ③ 11 ④ 12 ⑤ 13 ⑥ 27 ⑦ 49

● `kurasu_num`には全体の
クラス数が入る

答え イ ③

```

(1) Kurasumei = ["1A", "1B", "1C", "1D", "2A", "2B", "2C",
                 "2D", "3A", "3B", "3C", "3D"]
(2) Kekka = ["", "", "", "", "", "", "", "", "", "", "", ""]
(3) Tensu = [42, 24, 13, 27, 11, 49, 65, 67, 54, 60, 65, 3]
(4) kurasu_num = 
(5) ichiban = -1
(6) i = 0
(7) n = -1
(8) i を 0 から  まで 1 ずつ増やしながら繰り返す:
(9)   もし ichiban < Tensu[i] ならば:
(10)   |    = 
(11)   |   n = 
(12) Kekka[n] = "最優秀賞"
(13) 表示する (Kekka[n], "は", , "です")

```

図2 最優秀賞のクラスを選出するプログラム

完成したプログラムをAさんが実行したところ、画面には「最優秀賞は2Dです」と正しく表示された。

- 繰り返す回数は0から11まで
- 全体のクラス数が12なので-1

答え ③kurasu_num-1

```

(1) Kurasumei = ["1A", "1B", "1C", "1D", "2A", "2B", "2C",
                "2D", "3A", "3B", "3C", "3D"]
(2) Kekka = ["", "", "", "", "", "", "", "", "", "", "", "", ""]
(3) Tensu = [42, 24, 13, 27, 11, 49, 65, 67, 54, 60, 65, 3]
(4) kurasu_num = 
(5) ichiban = -1
(6) i = 0
(7) n = -1
(8) i を 0 から  まで 1 ずつ増やしながら繰り返す :
(9)   もし ichiban < Tensu[i] ならば :
(10)    = 
(11)    = 
(12) Kekka[n] = "最優秀賞"
(13) 表示する (Kekka[n], "は", , "です")

```

図2 最優秀賞のクラスを選出するプログラム

完成したプログラムをAさんが実行したところ、画面には「最優秀賞は2Dです」と正しく表示された。

- Tensu[i]を順番に見ていき
ichibanより大きければその値を
ichibanに入れる
- その際その地点で得点が高い
iをnに入れる

答え エ 0
オ Tensu[i]

答え カ i

```

(1) Kurasumei = ["1A", "1B", "1C", "1D", "2A", "2B", "2C",
                 "2D", "3A", "3B", "3C", "3D"]
(2) Kekka = ["", "", "", "", "", "", "", "", "", "", "", "", ""]
(3) Tensu = [42, 24, 13, 27, 11, 49, 65, 67, 54, 60, 65, 3]
(4) kurasu_num = 
(5) ichiban = -1
(6) i = 0
(7) n = -1
(8) i を 0 から  まで 1 ずつ増やしながら繰り返す :
(9)   もし ichiban < Tensu[i] ならば :
(10)   |    = 
(11)   |   n = 
(12) Kekka[n] = "最優秀賞"
(13) 表示する (Kekka[n], "は", , "です")

```

図2 最優秀賞のクラスを選出するプログラム

完成したプログラムをAさんが実行したところ、画面には「最優秀賞は2Dです」と正しく表示された。

- 一番得点が高い添字番号が n に入っている
- それをクラスに当てはめる

答え
キ ①kurasumei[n]

問3 ク・ケ・コ

7

- やりたいことは一旦全てのクラスに敢闘賞を当てはめる

0

1

2 . .

- Kekka=[“敢闘賞”、 “敢闘賞”、 “敢闘賞”]

- その過程でもし $ichiban < tensu[i]$ に当てはまるクラスがあれば
最優秀賞に書き換える

- まずIAは42点でichiban>-1より大きいので最優秀賞が代入される

0

1

2 . .

- Kekka=[“最優秀賞”、 “敢闘賞”、 “敢闘賞”]

- 次に42点を超えてくるのは49点の2Bなので最優秀賞が代入される
このとき IAの最優秀賞が消えて、代わりに敢闘賞が代入されなければならないがそのようなプログラムの記述はないのでIAの最優秀賞は残る
- 同じように65点の2Cは49点の2Bを超えるが2Cの最優秀賞は残る

答え

0 IA ⑤2B ⑥2C