Solution 14: Python Swap Numbers

1) The glasses idea (ordering)

Correct order:

1) Pour **Glass A** into **Temp**
2) Pour **Glass B** into **Glass A**
3) Pour **Temp** into **Glass B**

So the blanks are:

- 2 Pour **Glass B** into **Glass A**
- 1 Pour **Glass A** into **Temp**
- 3 Pour **Temp** into **Glass B**

2) Fill in the blanks: swap two numbers with `temp`

```
a = 6
b = 7

temp = a
a = b
b = temp
```

After the swap, `a` is **7** and `b` is **6**.

3) What is the output?

Step by step: - First print shows `a = 10 , b = 3` - After swapping, print shows `a = 3 , b = 10`

```
a = 10 , b = 3
a = 3 , b = 10
```

4) Debugging (2 bugs)

Corrected lines: - Line A: `temp = a` - Line B: `b = temp`

Correct full code (one correct fix):

```
a = 5
b = 9

temp = a
a = b
b = temp

print(a, b)
```

Output:

```
9 5
```

5) What does "in-place" mean?

- In-place means we **do** change the same list.
- We **don't** create a new list.

6) Trace one full reverse (5 letters)

Start: `["A", "B", "C", "D", "E"]`

- After 1st swap (index 0 and 4): `["E", "B", "C", "D", "A"]`

- After 2nd swap (index 1 and 3): `["E", "D", "C", "B", "A"]`

7) What is the output?

Reversing `[1, 2, 3, 4]` gives:

```
[4, 3, 2, 1]
```

8) Fill in the missing code (reverse a list)

One correct completion:

```
items = ["X", "Y", "Z", "W"]

left = 0
right = len(items) - 1

while left < right:
    temp = items[left]
    items[left] = items[right]
    items[right] = temp

    left = left + 1
    right = right - 1

print(items)
```

Output:

```
['W', 'Z', 'Y', 'X']
```

9) Write code: swap the first and last item only

One possible solution:

```
letters = ["A", "B", "C", "D", "E"]

temp = letters[0]
letters[0] = letters[len(letters) - 1]
letters[len(letters) - 1] = temp

print(letters)
```

Output:

```
['E', 'B', 'C', 'D', 'A']
```