1. First one
2. 10; ERROR; [10,20,20,30,30,'The End']; [10,5,30]; 40
3. 'I am Brown, I exist' ; 'I am White, I exist' ; I am Grey, I exist 'I hide' ; TypeError ; TypeError ; __main__.Animal
4. F
5. F
6. False?cuz order isn't guaranteed in a set so binary search wouldn't work reliably
7. F
8. F
9. Skip
10. True
11. True
12. True (ignore: after print)
13. F
14. skip/python tutor
15. 4th
16. None of the above
17. Skip
18a.

```
Shopping_list = {'potatoes': 3, 'carrots':2, 'eggs':3, 'peas': 30}
Stock = {'potatoes': 3, 'carrots':2, 'eggs':3, 'peas': 50, 'apples': 4}
Prices =  {'potatoes': 30, 'carrots':20, 'eggs': 5, 'peas': 4, 'apples': 40}
```

18b.

```
for key in shopping_list:
     if stock[key] < Shopping_list[key]:
        print("Not enough food in this store")
        return False
     return True
```

18c.

```
total = 0
for key in shopping_list:
     total += shopping_list[key] * prices[key]
to_return = total <=limit
return (to_return, total);
```

18d.

```
# assume you have `stock` defined outside function
def go_shopping(limit, shop_list):
    if not enough_ingredients+(shop_list):
```

```
        return False
    enough, total = enough_money(limit, shop_list)
    if not enough:
        return "Go back to work"
    else:
        return total
```

19. Skip
20. 'FinalExam()'
21. Skip
22. Skip
23.

If either stack is empty, return another one.
Create an empty stack `temp`
Loop until either stack1 or stack2 is empty.
On each iteration compare the tops and pop the smaller, then push it to `temp`

If one of the stacks is not empty yet, loop pop and push each element to temp.

Then loop again, pop, and push elements to one of the empty stacks to reverse an order.

24.

```
[lst[i] * i for i in range(0, len(lst), 2)]
```

25a. AttributeError: 'Final' object has no attribute 'x'
25b. x is an instance attribute of Exam, Final doesn't inherit it
26. Skip
27. Skip