

Facit och kommentarer till tenta 2023-08-22 i DA2005 och DA2004

Del A: flervalsfrågor

1. A
2. B
3. A
4. B
5. C
6. B
7. B
8. D
9. D
10. D

Del B: kodfrågor

11. Programmeringsspråk är i allmänhet dåliga på att representera decimaltal och att jämföra två flyttal för likhet, här $x + y$ och 0.3, kan lätt bli missvisande. Ett standardexempel är att det blir fel för $x = 0.1$ och $y = 0.2$.

12.

A. Det görs ett **return**, men inget värde returneras.

B. Exempellösning:

```
def interact(menu, valid_choices):
    '''
    Get an answer from a user.
    menu - a string presenting the choices.
    valid_choices - a list of strings containing the alternatives
    for the user.
    '''
    try:
        while True:
            print(menu)
            answer = input('What is your choice? ')
            if answer in valid_choices:
                return
            else:
                print('That is not a valid answer. Try again.')
    except EOFError:
        print('Bye!')
        sys.exit()
    except KeyboardInterrupt:
        print('Bye!')
        sys.exit()
```

13.

A. Funktionen räknar och returnerar antalet förekomster av värde i variabeln x i listan `lst`.

B. Exempellösning

```
def count_elems(lst, x):
    count = 0
    for elem in lst:
        if x == elem:
            count += 1
    return count
```

14.

A. Exempellösning:

```
def beauty_stats(butterfly_beauty):
    beauty_sum = {True: 0, False: 0}
    count = {True: 0, False: 0}
    for name, (hibernating, beauty) in butterfly_beauty.items():
        beauty_sum[hibernating] += beauty
        count[hibernating] += 1

    print('Hibernating Beauty')
    for hibernating in [True, False]:
        mean = beauty_sum[hibernating] / count[hibernating]
        print(f'{hibernating!s:12} {mean}')
```

B. Exempellösning:

```
def keep_perfect(butterfly_beauty):
    predicate = lambda b: butterfly_beauty[b][1] == 10
    transform = lambda b: (b, butterfly_beauty[b][0])
    return list(map(transform, filter(predicate, butterfly_beauty)))
```

C. Exempellösning:

```
class Butterfly:
    def __init__(self, speciesname, beauty, hibernating):
        self.speciesname = speciesname
        self.beauty = beauty
        self.hibernating = hibernating

insect1 = Butterfly('Sorgmantel', 9, True)
```