

# Facit och kommentarer till prov 2019-04-17 i DA2004

## Del A: flervalfrågor

1. *d*
2. *e*
3. *d*
4. *a*
5. *c*
6. *a, b, c*
7. *a, b*
8. *b*
9. *a och b*
10. *d*

## Del B: kodfrågor

11. Ett exempel är 'abba'. Funktionen avgör om en sträng är ett palindrom eller inte.

12. Möjlig lösning:

```
def dropwhile(pred, lst):  
    i=0  
    while pred(lst[i]):  
        i += 1  
    return lst[i:] # Slicing gives a copy
```

Notera att vi returnar en kopia av (del-)listan, så vi manipulerar inte lst.  
Funktionen finns redan i standardmodulen itertools.

13. Exempellösning:

```
def mul_tabell():  
    while True:  
        try:  
            s = input('Vilken multiplikationstabell?_')  
            if s == '':  
                return  
            n = int(s)  
            break  
        except:  
            print('Bad_input,_try_again_or_hit_return_to_quit.')
```

```
for i in range(1,13):  
    print(s, '*',i,'=',i*int(s))
```

14. Om man vill programmera defensivt är instruktionen `assert` bra att komma ihåg!

```

def polynomial_degree(p):
    assert p != []
    return len(p) - 1

```

15. Problemet är att man kan komma att returnera uppslagstabeller som inte har alla fyra nukleotiderna A, C, G, och T inlagda som nycklar. Tex skulle anropet `count_nucleotides('AAA')` returnera `{'A':3}`, och det är inte enligt specifikation.

16. Exempellösning:

```

def rootfinder_fixer(f):
    a, b = rootfinder(f)
    solution = a
    if b:
        if abs(b) > abs(a):
            solution = b
    return solution

```

17. Exempel, med metoden för fråga 19 inlagd.

```

class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

    def compare(self, other):
        return self.x == other.x and self.y == other.y

```

18. `p1 = Point(1,2)`  
`p2 = Point(3,4)`

19. Se 17 ovan.

20. `if p1.compare(p2):`  
`print('Klar!')`

Överkurs: Om man definierar en metod `__eq__`, som så här,

```

def __eq__(self, other):
    return self.x == other.x and self.y == other.y

```

i sin klass så kan man använda jämförelseoperatorn istället:

```

if p1 == p2:
    print('Klar!')

```