Facit och kommentarer till tentamen 2023-03-14 i DA4003

Del 1: flervalsfrågor (1p per fråga)

1. B, D

2. B

3. E

4. A

5. D

6. B, E

7. A

8. C, D, E

Del 2: kodfrågor

9. (a) Möjlig lösning:

```
int sum_numbers(int* p, int n) {

  int sum = 0;

  for (int i = 0; i < n; i++)
    sum += *(p + i);

  return sum;

}
```

   (b) Möjlig lösning:

```
int sum_numbers(int* p, int n) {

  int sum = 0;
  int i = 0;

 start:
  sum += *(p + i);
  i++;
  if (i < n) {
    goto start;
  }

  return sum;

}
```

10. (a) Möjlig lösning:

```
class Vehicle {

    public String manufacturer;
    public int number_of_wheels;
    private String owner;

    public Vehicle(String m, int n, String o) {
        manufacturer = m;
```

```
            number_of_wheels = n;
            owner = o;
        }

        public String get_owner() {
            return owner;
        }
    }
```

(b) Möjlig lösning:

```
class Car extends Vehicle {
    public int km;

    public Car(String m, String o, int k) {
        super(m,4,o);
        km = k;
    }
}
```

(c) Möjlig lösning:

```
Car new_tesla = new Car("Tesla","Elon Musk",0);
```

11. (a) Möjlig lösning:

```
partition :: (a -> Bool) -> [a] -> ([a],[a])
partition p [] = ([],[])
partition p (x:xs) =
  let (ys,zs) = partition p xs
  in if p x then (x : ys,zs) else (ys,x : zs)
```

(b) Möjliga lösningar (alternativ 2 och 3 ger 3p, men alternativ 1 ger 2p pga Eq kravet i typen):

```
-- Alternativ 1 (ok trots Eq i typen)
all :: Eq a => (a -> Bool) -> [a] -> Bool
all p xs = [] == snd (partition p xs)

-- Alternativ 2 (utan Eq a kravet)
all :: (a -> Bool) -> [a] -> Bool
all p xs = isEmpty (snd (partition p xs))
  where
    isEmpty [] = True
    isEmpty _  = False

-- Alternativ 3 mha null
all :: (a -> Bool) -> [a] -> Bool
all p xs = null (snd (partition p xs))
```

12. (a) Möjlig lösning:

```
tribonacci(0,0).
tribonacci(1,0).
tribonacci(2,1).
tribonacci(N,T) :-
    N1 is N - 1,
    N2 is N - 2,
    N3 is N - 3,
    tribonacci(N1,T1),
    tribonacci(N2,T2),
    tribonacci(N3,T3),
    T is T1 + T2 + T3.
```

(b) Möjlig lösning:

```
zip([],_,[]) :- !.
zip(_,[],[]) :- !.
zip([X | XS],[Y | YS],[(X,Y) | ZS]) :-
    zip(XS,YS,ZS).
```