**Kaj sem se na vajah naučil:**

Na vajah sem spoznal uporabo oziroma osnovne metode z delo z drevesi. Tako smo s pomočjo teh razvili nekaj preprostih funkcij za pridobivanje nekih lastnosti iz dreves.

**Ideje in pristopi:**

Uporabljal sem osnovne metode, ki jih pozna podatkovna struktura drevo. Ideje sem vedno premišljeval rekurzivno in tako rešil večino. Sprva sem si narisal neko drevo in po tej v mislih pognal algoritem in tako oblikoval rekurzivni if stavek in pa premislil kaj more funkcija vrniti.

**Opažanja in komentarji:**

Pri vajah sem spoznal, da z drevesi najlažje delamo z rekurzijo in da se vedno spomnimo definicije drevesa. Naloge so mi bile všeč, saj si potreboval dober premislek in potem ko si si zastavil problem si hitro rešil nalogo.

**Par:**

V paru sem bil z Markom Klemenškom.

**Sestavljanje testnih primerov in reševanje naloge:**

Najprej sva si v paru zbrala naloge, ki jih bova rešila in si jih razdelila. Nato sva si narisala nekaj dreves in tako izoblikovala različne vrste dreves (izrojeno, polno) in tako izoblikovala testne primere. Ideja testnih primerov je bila, da preizkusiva najine funkcije na različnih vrstah dreves in tudi na praznem.

Potem sva se lotila reševanja, najprej sem si narisal kazalce, kako bi moral program začeti in kaj naredi v primeru ko pride do lista v drevesu. Tako sem dobil osnovno idejo in s tem zapisal kodo.

**Dodatne dogovorjene zahteve:**

Dodatno sva se dogovorila, da je tudi prazno drevo izrojeno drevo, zato funkcija v primeru, da za parameter podamo prazno drevo vrne True.

**Koda posameznih programov:**

V navodilih piše, da dodamo tudi kodo programov. Sem pa dodal tudi vso kodo zraven v .rar datoteki, saj je gledanje v kodo v Wordu boleče. Prav tako sem izpustil vse teste in so le ti dodani v datotekah. Tako, da so tukaj dodane samo funkcije, ki rešijo dane probleme.

1. **Vrni višino drevesa**

def vrni\_visino\_drevesa(drevo):  
 *'''Vrne višino drevesa 'drevo'.'''* if drevo.prazno():  
 return 0  
 return max(1 + vrni\_visino\_drevesa(drevo.levoPoddrevo()), 1 + vrni\_visino\_drevesa(drevo.desnoPoddrevo()))

1. **Vrni število vseh vozlišč**

def stevilo\_vseh\_vozlisc(drevo):  
 *"""Vrne stevilo vseh vozlisc v drevesu"""* if drevo.prazno():  
 return 0  
 return 1 + stevilo\_vseh\_vozlisc(drevo.levoPoddrevo()) + stevilo\_vseh\_vozlisc(drevo.desnoPoddrevo())

**5. Ali je dano drevo izrojeno**

def ali\_je\_izrojeno(drevo):  
 *"""Vrne true, ce je dvojisko drevo izrojeno, dogovor: prazno drevo je izrojeno drevo"""* if drevo.prazno():  
 return True  
 if not drevo.levoPoddrevo().prazno() and not drevo.desnoPoddrevo().prazno():  
 return False  
 return ali\_je\_izrojeno(drevo.levoPoddrevo()) and ali\_je\_izrojeno(drevo.desnoPoddrevo())

**12. Vrni tabelo vseh podatkov, ki so v dvojiškem drevesu na nivoju k**

def kti\_nivo(drevo, k):  
 *"""Vrne vse elemente v ktem nivoju drevesa"""* if drevo.prazno():  
 return []  
 if k == 1:  
 return [drevo.vrniPodatek()] + kti\_nivo(drevo.levoPoddrevo(), k - 1) +\  
 kti\_nivo(drevo.desnoPoddrevo(), k - 1)  
 return kti\_nivo(drevo.levoPoddrevo(), k - 1) + kti\_nivo(drevo.desnoPoddrevo(), k - 1)