

Cvičení z Algoritmizace a Programování 1

10. cvičení

PŘÁTELSKÉ PŘIPOMENUTÍ:

Podmínky zápočtu – Algoritmizace

- domácí úkoly – celkem 8 (každý za bodů), potřebujete získat 70 % bodů
 - navíc, za každou účast na cvičení dostanete 1 bod -> celkem až 14 bonusových bodů
- předvedení řešení (alespoň) 1 úkolu u tabule
- zápočet nutný ke zkoušce -> doporučuji se ho snažit získat co nejdříve

Prohledávání – obecný iterativní algoritmus verze 1

```
objevene = {pocatecni_stav}
k_prozkoumani = (pocatecni_stav)
while k_prozkoumani:
    stav = k_prozkoumani.get()
    for naslednik in naslednici(stav):
        if naslednik not in objevene:
            objevene.add(naslednik)
            k_prozkoumani.add(naslednik)
```

Prohledávání – obecný iterativní algoritmus verze 2

```
prozkoumane = {}  
k_prozkoumani = (pocatecni_stav)  
while k_prozkoumani:  
    stav = k_prozkoumani.get()  
    if stav not in prozkoumane:  
        prozkoumane.add(stav)  
        for naslednik in naslednici(stav):  
            k_prozkoumani.add(naslednik)
```

(10 bodů) Je zadán binární strom. Navrhněte efektivní algoritmus, který zjistí **průměrnou výšku stromu**, definovanou jako

součet délek všech cest z kořene do listu / počet listů .

(a) Svoje řešení zapište jako **funkci v Pythonu** (která obdrží kořen binárního stromu a vrátí jeho průměrnou výšku), přitom využijte definici třídy pro vrchol binárního stromu níže a kód opatřete **komentáři**,

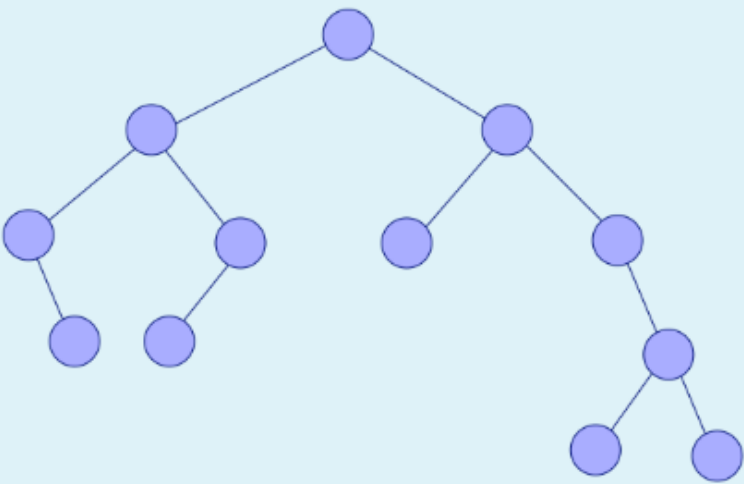
(b) zdůvodněte **správnost**,

(c) odvodte **časovou a prostorovou složitost**.

```
class VrcholBinStromu:  
    """třída pro reprezentaci vrcholu binárního stromu"""  
    def __init__(self, x = None, levy = None, pravy = None)  
        self.info = x      # data  
        self.levy = levy   # levé dítě  
        self.pravy = pravy # pravé dítě
```

Příklad:

Vstup:



Výstup:

3.2
(zdůvodnění: $(3+3+2+4+4) / 5 = 3.2$)

Práce se soubory

- Viz `Files.py`

Načítání stromu ze souboru

- Soubor na webu (strom.txt)
- Formát:
 - Každý vrchol má unikátní identifikátor (typu string) a hodnotu (int)
 - 1. řádek: identifikátor kořene
 - Další řádky:
 - 1 řádek = 1 vrchol stromu, v libovolném pořadí!
 - Formát řádku:
 - `id -> hodnota, id_levého_dítěte, id_pravého_dítěte`
 - id dětí mohou být prázdná (= vrchol dané dítě nemá)
 - Např. `n1 -> 5, ,n2` – vrchol „n1“ má hodnotu 5, levé dítě nemá, jeho pravé dítě je vrchol „n2“