

IB111 – cv. 8

Abstraktní datové typy

Miroslav Kadlec

Obsah

- Seznam
- Zásobník
 - Interpretace postfixového výrazu
- Fronta
- Slovník
 - Morseovka
- Množina

Seznam - teoretičtěji

- Implementace seznamu:
 - Jednosměrně vázaný seznam
 - Obousměrně vázaný seznam
 - Dynamické pole
- V Pythonu?
 - A jak bychom to ověřili?
- Již jsme zkoušeli
 - Append, len, copy, deepcopy, min, max, slicing
- Další vychytávky
 - Insert, del, reverse, sort

Zásobník

- Datová struktura pro ukládání podle principu LIFO
- Operace charakteristické pro zásobník:
 - `stack.push(x)` `#vlozi x na vrchol`
 - `x = stack.pop()` `#vrati prvek z vrcholu a smaze ho`
 - `x = stack.empty()` `#vrati true, pokud je prazdny`
 - `(x = stack.top())` `#vrati prvek z vrcholu a necha ho tam`
- Využití
 - Kontola závorkování, ukládání kontextu v procesoru, analýza postfixového výrazu
- V Pythonu – můžeme implementovat seznamem
 - Co funkce "push", "empty" a "top"?

Fronta

- ADT pro ukládání podle principu FIFO
- Charakteristické funkce
 - Opět push a pop, jen na jiné části struktury
- Využití
 - komunikace mezi procesy v Unixu, vyrovnávací paměť pro datové toky
- V Pythonu
 - Seznamem by šla, existuje zvláštní implementace
 - from collections import **deque**
 - Viz sbírka <https://www.fi.muni.cz/IB111/sbirka/>

Slovník

- ADT pro ukládání hodnot pod klíče namísto pod indexy
- Využití:
 - Kódování, uchovávání hodnoty pro různé lidi, přístroje, ...
- slovník = {"klic":66, "jinyKlic":0, "tretiKlic":0}
 - Klíč musí být neměnný – jak je na tom řetězec a seznam?
- Úkol - morseovka

Fronta

- ADT pro ukládání podle principu FIFO
- Charakteristické funkce
 - Opět push a pop, jen na jiné části struktury
- Využití
 - komunikace mezi procesy v Unixu, vyrovnávací paměť pro datové toky
- V Pythonu
 - Seznamem by šla, ale existuje zvláštní implementace
 - from collections import **deque**
 - Viz sbírka <https://www.fi.muni.cz/IB111/sbirka/>

Množina

- ADT pro ukládání bez možnosti přístupu ke konkrétnímu prvku
- Charakteristické funkce
 - Add
 - Remove
 - Size
- V Pythonu
 - `my_set = set()` `my_set.add(4)`
 - `my_set.remove(4)` `my_set.update([1, 3])`
- Využití:
 - Kontrola unikátnosti, brainstorming