

Programování II (NPRM045) – Cvičení 4

Obsah cvičení

- jednodušší spojové struktury

Příklady

1. Písemka: Napište jednoduchou implementaci zásobníku pomocí dynamicky alokované paměti.
2. Implementujte proceduru, která vypíše všechny prvky jednosměrného lineárního spojového seznamu.
3. Zkuste vlastními slovy popsat, co je to přesně konstanta `nil` a k čemu se používá.
4. Příkaz `dispose` slouží k uvolnění bloku paměti. Co to přesně znamená a jak prvky přesně odstranit, abychom datovou strukturu udrželi v konzistentním stavu? Kam ukazuje ukazatel po provedení `dispose`? Co když na uvolňovaný blok ukazuje více ukazatelů? Je pak třeba `dispose` volat vícekrát?
5. Implementujte funkci `otocNedestruktivne(var prvniPrvek: UkPrvek): UkPrvek`, která vytvoří nový seznam, který bude obsahovat prvky zadaného seznamu v obráceném pořadí.
6. Implementujte proceduru `pridejNaKonec(var prvniPrvek: UkPrvek; co: integer);`, která přidá za poslední prvek seznamu hodnotu `co`.
7. Implementujte proceduru `odstranPosledni(var prvniPrvek: UkPrvek);`, která odstraní ze seznamu poslední prvek (tj. doskáče od prvního prvku až po předposlední a předposlednímu odtrhne následovníka). Technický problém: jednoprvkový seznam - je třeba zvlášť ošetřit případ, kdy má seznam právě jeden prvek.
8. Kdy bude uvolněna paměť, kterou zapomeneme odalokovat?
9. Napište proceduru, která zdvojnásobí všechny prvky spojového seznamu. Příklad: bude-li na vstupu `[5, 2, 8, 3]`, po úpravě musí seznam vypadat takto: `[5, 5, 2, 2, 8, 8, 3, 3]`.

Domácí úkol

1. Implementujte funkci `velikostSeznamu(var prvniPrvek: UkPrvek): Integer`, která vrátí počet prvků seznamu.
2. Implementujte funkci `vypisPosledni(var prvniPrvek: UkPrvek): Integer;`, která bere ukazatel na první prvek a vrací hodnotu posledního prvku.
3. Napište proceduru, která otočí zadaný spojový seznam, aniž by používala procedury `new` a `dispose` (tzv. nedestruktivní otočení). Cílem je tedy přepojit ukazatele tak, aby mířily opačně, a za nový začátek seznamu prohlásit jeho původní konec. Pozor! Toto je těžká úloha. Nejprve si vše v klidu promyslete (nejlépe s použitím tužky a gumy) a teprve pak pište kód.
4. Poctivě si projděte všechny animace na Krylově stránce.