
Dinamikus rendezett minta törléssel

Dinamikus rendezett minta műveletein kívül adat szerinti és pozíció szerinti törlést is végezhetünk:

Feladat

Írjunk olyan programot, amely megvalósítja az alábbi műveletet!

- **Adat(x)**: az x paraméterben kapott elemet hozzáveszi a halmazhoz. Ha az x elem már korábban nem szerepelhetett Adat(x) műveletben, akkor a ez a művelet hatástalan.
- **Elemszam**: Az eddig beérkezett adatelemek számát adja.
- **Kadik(k)**: Az eddig beérkezett számok közül a rendezés szerinti k-adikat adja eredményül.
- **Hanyadik(x)**: Ha az x adat nem szerepelt eddig Adat(x) műveletben, akkor az eredménye 0, egyébként az x elem rendezésbeli sorsszámát.
- **KTorol(k)**: Az eddig beérkezett adatelemek közül a rendezés szerinti k-adikat törli a halmazból.
- **Torol(x)**: Az eddig beérkezett adatelemek közül az x elemet törli a halmazból.

Megvalósítás

Pascal program esetén

```
uses rendninta;
```

Műveletek Pascal deklarációja

```
procedure Adat(x:longint);  
function Elemszam: longint  
function Kadik(k:longint) : longint;  
function Hanyadik(x:longint) : longint;  
procedure KTorol(k:longint);  
procedure Torol(x:longint);
```

A műveletek C/C++ deklarációja

```
#include "rendminta.h"  
void Adat(int x);  
int Elemszam();  
int Kadik(int k);  
int Hanyadik(int x);  
void KTorol(int k);  
void Torol(int x);
```

Gyakorlás

A minta.zip fájlban letölthető egy üres minta.

Korlátozások

A számok értéke legfeljebb 2000 000 000.

Az Adat és a Kadik és Hanyadik függvényeket legfeljebb 100000-szer hívják.

Időlimit: 0.3 mp.

Memórialimit: 64 MiB

A tesztek 30%-ában a számok értéke <10000.

A programod nem írhat és nem olvashat semmilyen állományt, a standard outputra sem írhat!