

Programovanie (1) v C/C++ 2024/25

Cvičenia 7, príklad 3

Pridávanie

Na prednáške 12 sme videli dynamické pole, ktoré pri pridávaní prvkov rastie podľa potreby. Keď počet prvkov v poli má prekročiť jeho alokovanú veľkosť, alokuje sa nové pole dvojnásobnej veľkosti, pôvodné prvky sa tam prekopírujú a nakoniec sa pridá nový prvok.

V tomto príklade máte rozšíriť dynamické pole o možnosť pridať naraz k prvkov (k kópií prvku x). Ak by sme ich pridávali po jednom, mohlo by sa zbytočne niekoľkokrát realokovávať pole. Namiesto toho si najskôr spočítame, aká by bola finálna veľkosť pola po pridávaní prvkov po jednom a potom zavoláme funkciu `reallocate` iba raz. Výpočet robíme tak, že aktuálnu veľkosť pola opakovane násobíme dvoma až kým nedosiahneme veľkosť dostatočnú na uloženie nových aj pôvodných prvkov. Napríklad sme ak v poli sme mali 2 prvky a chceme pridať ďalších 12, potrebujeme mať v poli miesto na aspoň 14 prvkov. Ak počiatočná veľkosť pola bola 5, po prvom vynásobení veľkosti dvoma dostaneme veľkosť 10, čo je ešte málo. Po druhom vynásobení dostaneme 20, čo bude veľkosť, ktorú pošleme do funkcie `reallocate`.

Priložená kostra obsahuje kód pre dynamické pole z prednášky s dvoma zmenami. Funkcia `init` dostane aj počiatočnú veľkosť pola, ktorá nebude vždy 2. Funkcia `reallocate` vypíše novú veľkosť na konzolu. Vašou úlohou je naprogramovať funkciu `addMany`, pričom dodržujte pokyny uvedené v zadaní a v hlavičke funkcie. Ostatné časti programu nemeňte.

Funkcia `main` načítava zo vstupu príkazy `init`, `add`, `addMany`, `get` a `set` a vypisuje ich výsledky (viď príklad vstupu a výstupu). Príkaz `print` vypíše obsah pola a `end` ukončí činnosť programu. Prvý príkaz bude vždy príkaz `init`. Načítanie a výpis je už v programe hotový, nemeňte ich.

Upozornenie: Odovzdané riešenia budú dodatočne kontrolované. Ak vaše riešenie nebude spĺňať podmienky zadania (napr. nepoužije uvedenú kostru), body za túto úlohu vám budú odobrané.

Príklad vstupu:

```
init 5
add 30
add 40
addMany 12 50
print
addMany 4 60
addMany 2 70
addMany 1 80
print
get 0
set 0 90
get 0
end
```

Príklad výstupu:

```
Inializing size 5
Adding value 30
Adding value 40
Adding value 50 number of copies 12
reallocate from 5 to 20
Printig all values
30 40 50 50 50 50 50 50 50 50 50 50 50
Adding value 60 number of copies 4
Adding value 70 number of copies 2
Adding value 80 number of copies 1
reallocate from 20 to 40
Printig all values
30 40 50 50 50 50 50 50 50 50 50 50 50 60 60 60 60 70 70 80
Value at position 0 is 30
Setting value at position 0 to 90
Value at position 0 is 90
Finishing
```