

2021 Aplikačné program... / Cvičenia



4 Trieda BigNum



Vytvoril/a Peter Koscelanský [Administrator]

Posledná aktualizácia: Nov 22, 2021

Implementujte triedu pre reprezentáciu celého čísla s ľubovoľnou presnosťou. Implementáciu samozrejme nechávam na vás, no nejde mi o efektívnosť implementácie, za to budú až bonusové body. Názov triedy bude **BigNum** a podporovať nasledovné operácie. Stiahnite si hlavičkový súbor a doplňte do neho implementáciu. Potom odovzdajte iba tento súbor (hlavne tam nedávajte žiadnu funkciu `main`).

Konštruktory

- Konštrukcia bez parametrov (inicializuje číslo na `0`)
- Konštrukcia z `int64_t`
- Konštrukcia z reťazca (`const std::string&`), ak reťazec nebude v číselnom formáte (malo by to vedieť spracovať čísla ako `-123`, `+0`, `0000`, `-00012`, ...), vyhodte výnimku. Biele znaky na začiatku a na konci neakceptujeme, teda `456` nie je dobrý vstup.

Podpora kopírovania

- Kopírovací konštruktor a kopírovací `operator=` (ak vám stačia automaticky generované, tak nemusíte definovať)

Unárne operátory

- Unárny operátor `+` ako vo výraze `b = +a;`
- Unárny operátor `-` ako vo výraze `b = -a;`

Binárne aritmetické operátory

- Aritmetické operácie `-`, `+`, `*`

- Ich `+=`, `*=`, `-=` ekvivalenty
- Na násobenie prosím použite trochu lepší algoritmus, ako n-krát spočítam čísla. Stačí aj také násobenie ako učia na základnej škole.

Relačné operátory

- Relačné operátory (`<`, `>`, `<=`, `>=`, `==`, `!=`), pri tomto vám stačí naimplementovať operátor `<=>`, dôležité je aby fungovali výrazy ako `a < b`, `a == b`...

Podpora streamov

- Podporu pre výstup do streamu, teda preťaženie operátora `<<`

Bonusové body

- (2body) Implementáciu `/`, `/=`, `%` a `%=` (celočíselné delenie) (<http://stackoverflow.com/a/5387432>)
- (1bod) Podpora operátora `>>` pre vstup zo streamov, vyextrahuje číslo zo streamu pokiaľ sa dá, nastaví `failbit` ak sa nedá extrahovať ani jeden znak, prípadne ak je na vstupe iba `-`, vtedy skonzumujeme `-` a nastavíme `failbit`. Tu si pozrite ako sa správa `int` a operátor `>>`, malo by sa to správať rovnako aj v našej triede. Musíte vyriešiť ako sa správajú neplatné vstupy, napríklad `xyz` a `-xyz`.
- (1bod) Rýchlosť, tu urobíme pár testov, ktoré testujú násobenie, sčítanie a odčítanie a prvých 5 implementácií získa ďalší bod. Čo urobiť aby vaša implementácia bola rýchla?
 - Implementácia (ale správna) Karatsubovho násobenia (pre veľké čísla, presne si to treba otestovať, ale povedzme väčšie ako cca. 10^{100}).
 - Ukladanie čísla nie po 10tkových cifrách, ale po väčších častiach (povedzme `int/2`)

Ak sa rozhodnete implementovať bonusy, tak príslušné makrá v hlavičkovom súbore nastavte na 1.



Odpovedať • Páči sa mi to

Odpovedať • Páči sa mi to

Odpovedať • Páči sa mi to

Odpovedať • Páči sa mi to

Odpovedať • Páči sa mi to

Odpovedať • Páči sa mi to

1. Pride prazdny string, ma byt bignum == 0?
2. Mozeme pouzivat uint8_t?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Nov 26, 2021

1. Moze byt, teste minuly rok brani aj prazdny string vyjimka aj 0,
2. Tomu nerozumieme, akoze v kode? To jasne mozes.

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

Hej len som dakde cital ze to moze robit sarapatu ale som neporozumel ze co.

Odpovedať • Páči sa mi to



Anonymný Nov 26, 2021

Zdravím,

chcel by som sa opýtať na vyhadzovanie výnimiek pri konštruktoroch, keďže dopĺňame len hlavičkový súbor, je dostatočné tam dať iba throw a try a catch nemame riešiť, pretože bude napísaný v testoch ale sa to rieši nejako inak?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Nov 26, 2021

Presne tak, vy rieste iba throw.

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

Co presne maju robit unarne operatory, kedze nemaju ziadne vstupne argumenty?

Ich deklaracia v headeri je:

```
1 const BigNum& operator+() const;
2 BigNum operator-() const;
```

Som z toho vcelku zmateny kedze tie funkcie su const.

Co presne znamenaju tieto vyrazy zo zadania:

```
1 b = +a;
2 b = -a;
```

```
2 0 = -a,
```

Znamená to toto:

```
1 BigNum(8) = +BigNum(7)
2 BigNum(8) = -BigNum(7)
```

alebo toto:

```
1 BigNum(8) = +(int)7
2 BigNum(8) = -(int)7
```

prípadne to znamená niečo iné?

A čo to má robiť v jednotlivých prípadoch?

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

Podľa mňa to má skonštruovať objekt podľa nejakého iného objektu. Napr.

```
1 BigNum(7)
2
3 BigNum2 = +BigNum
4 BigNum3 = -BigNum
```

A výsledkom bude, že BigNum2 bude 7 a BigNum3 bude -7.

Je to správne?

A ešte by som sa spýtal či treba riešiť to, že user zadá to konštruktora cez int64 väčšie číslo ako max hodnotu. Máme to nejak ošetriť (výnimka) alebo nie?

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

Ja len k tomu int64. Nemáš ako zistiť, že používateľ zadal väčšie číslo, lebo ak by sa mu to aj nejak podarilo 😊, tak hodnota pretečie a ty aj tak dostaneš len nejakú int64 hodnotu, podľa ktorej skonštruješ BigNum.

Odpovedať • Páči sa mi to

Peter Koscelanský [Administrator] Nov 27, 2021



Tak to už by sme nejak museli hackovať C++ (použiť nejaký __int128).

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Nov 27, 2021

Unarný + vlastne ma iba vrátiť kópiu samého seba.

Unárny - má vrátiť kópiu v ktorej bude invertované znamienko.

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

Môžem sa spýtať ako sú robené tie test casey, keď mi tam padá prosté from int a from string? :D
aj keď realne mi ich to načítava normálne... sú nejaké určité casey na ktoré sa mám zamerať,
alebo ktoré by ma len tak nenapadli??

Odpovedať • Páči sa mi to



Anonymný Nov 27, 2021

A ešte, že koľko máme testovacích pokusov na toto zadanie?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Nov 28, 2021

3

Odpovedať • Páči sa mi to



Anonymný Nov 28, 2021

Ja som osetroval iba to, čo je napísané v zadani a preslo mi to. Skús otestovať pre string:

+123, 123, -123

+000123, 000123, -000123

+0000, 0000, -0000

-0, 0, +0

^^^ (všetko validne)

+123w, 123w, -123w

lw123, w123, lw123

`+w123, w123, -w123``+1w23, 1w23, -1w23``+0w00123, 0w00123, -0w00123``+00w00, 00w00, -00w00``+0000w, 0000w, -0000w`

^^^ (nevalidne)

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

No akoze všetky tieto vstupy mi presli, dalsia vec co nechapem ako mi nemoze prejsť from int ... ved tam len robis modula, delis a zapisujes cifry... tam mas nejake vstupy ?

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

Skontroluj si ci ti spravne funguje modulo na zaporne cisla.

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

A tipujem, ze nuly si odstranoval z tych vstupov, cize 000123 ti ulozi ako 123 atd a este nulu si ulozil vzdy ako 0, bez znamienka ze?

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Nov 28, 2021

Ak chcete pozrieť, ci tam mate naozaj chybu, tak mi mozete napisat na teams a ja vam to pozriem, ale testy su dobre.

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

Ked mam pretazenie operatora << spravene tak ze to vypise okrem cisla aj new line, moze vypis new line sposobit, ze mi neprejdú testy?

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

Asi ano, << musi vypisat len hodnotu, nic naviac

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Nov 28, 2021

Presne tak, preco by tam mal byt newline, to je totalne proti filozofii 😊

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

Chcel by som sa spýtať či by mal program pokračovať po vyhodení výnimky alebo stopnúť sa ?

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Nov 29, 2021

Ty nic, kedze implementujes len tuto triedu, tak ti to je jedno.

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021

Zdravim,

ako presne maju fungovat binarne +, - a *?

mame vratat lhs alebo novy objekt?

Odpovedať • Páči sa mi to

**Peter Koscelanský [Administrator]** Nov 29, 2021

Sice, neskoro ale odpoviem. n headri maju return value BigNum, takže v podstate nemozu vrat nic ine ako novy objekt, to by museli mat navratovu hodnotu BigNum&

Odpovedať • Páči sa mi to

**Anonymný** Nov 28, 2021 [↗](#)

V hlavickovom subore je napisane ze nemame menit public interface. Mozeme donho pridavat nove operatory a funkcie?

Odpovedať • Páči sa mi to



Peter Koscelanský [Administrator] Nov 29, 2021

To som viackrát hovoril, ano pridať môžete.

Odpovedať • Páči sa mi to