

# Контролно

Ще проведем контролното в компютърни кабинети 107, 306, 309, 314, 321 ФМИ.

В тази задача трябва да реализирате специален калкулатор. Той трябва да поддържа няколко паралелни калкулации. С всяка от тях трябва да може да се изпълняват определени операции. Трябва да може да се показва и текущото състояние на всички калкулации.

Всяка калкулация съдържа редица от реални числа – нейните аргументи. Максималната дължина на такава редица е 255. Няма ограничение на броя калкулации, които потребителят може да създава.

Вашата програма трябва да може да изпълнява следните команди:

1. CREATE – добавя нова калкулация в края на списъка от текущите.
2. PUSH <NUMBER> <CINDEX> – добавя реалното число <NUMBER> в калкулацията със индекс <CINDEX> (цяло неотрицателно число).
3. POP <CINDEX> – премахва последния аргумент от калкулацията с индекс <CINDEX>. Ако тя остане без аргументи, трябва да се премахне от списъка с калкулации. В такъв случай индексите на тези след нея се променят като намаляват с единица.
4. ADD – за всяка калкулация премахва последните ѝ 2 аргумента, събира ги и записва резултата обратно, като нов аргумент за калкулацията. Ако дадена калкулация има по-малко от два аргумента, тя не се променя.
5. SUBTRACT -- работи аналогично на ADD, но изважда предпоследния от последния аргумент, вместо да ги събира.
6. NEGATIVE – обръща знака на последния аргумент на всяка калкулация.
7. MAX – Извежда на екрана най-големия аргумент на всяка от калкулациите. Ако дадена калкулация няма елементи, за нея да се изведе текст "none"
8. PRINT – принтира индекса и аргументите на всяка сметка на отделен ред.
9. EXIT – изход от програмата.

Операциите CREATE и PUSH трябва да работят със сложност  $O(1)$ . POP трябва да работи със сложност  $O(1)$ , освен когато се наложи да премахне калкулация. В такъв случай тя може да работи със сложност  $O(N)$ , където  $N$  е броят на калкулациите в програмата. ADD, SUBTRACT, NEGATIVE, MAX и PRINT трябва да работят със сложност  $O(N)$ .

Трябва сами да реализирате всички алгоритми и структури от данни, които използвате в програмата. НЕ Е разрешено да се използват стандартни имплементации, като например тези в STL.

Ако искате, можете да използвате класа `std::string`.

Пример за работата на програмата:

```
> CREATE
> PRINT
0:[]
> CREATE
> PRINT
0:[]
1:[]
> MAX
0: none
1: none
> PUSH 1 0
> PUSH 3 0
> PUSH 7 1
> PRINT
0:[1,3]
1:[7]
> MAX
0: 3
1: 7
> ADD
> NEGATIVE
> PRINT
0:[-4]
1:[-7]
> CREATE
> PUSH 10 1
> SUBSTRACT
> PRINT
0:[-4]
1:[17]
> POP 0
> PRINT
0:[17]
> EXIT
```