

Изпит по Функционално програмиране

спец. Информатика, Компютърни науки, 1 поток, и Софтуерно инженерство

14.02.2020 г.

Вариант А

Задача 1. База данни се описва с име (низ) и размер в MB (цяло число). Сървър за данни се описва с име (низ), капацитет в MB (цяло число) и списък от бази данни. Нека е даден списък от сървъри **I**.

а) (8 т.) Да се реализира функция **maxFree**, която намира името на този сървър в **I**, който има най-много свободно място (разликата между капацитета на сървъра и сумата на размерите на базите данни).

б) (12 т.) Да се реализира функция **tryRemove**, която по име на сървър **s** проверява дали е възможно всички бази от данни от **s** да се прехвърлят по другите сървъри от **I** (по не повече от една база на сървър) така, че **s** да бъде изключен. Функцията да връща новия списък от сървъри след прехвърлянето, ако е възможно, или оригиналния списък **I**, ако не е възможно.

Упътване: достатъчно е да се подредят сървърите и базите данни съответно по свободно място и размер и да се провери дали има съответствие. Ако такова няма, значи разпределение не е възможно.

в) (Бонус 15 т.) Да се реши б) без ограничението “не повече от една база на сървър”. Не е нужно решението да е оптимално по време.

Задача 2. (15 т.) Да се реализира функция **comps**, която по даден списък от едноместни числови функции **fl** генерира безкраен поток от всички възможни композиции на функциите във **fl**, включително и празната композиция (функцията идентитет).

Пример: `comps [f, g] → [id, f, g, f.f, f.g, g.f, g.g, ...]`

Задача 3. (15 т.) Разглеждаме дървета от числа с произволен брой наследници на всеки възел, с представяне по ваш избор. Числата в дървото могат да се повтарят. Да се реализира функция **minPredecessor**, която по дадено дърво и число **x** намира най-малкото число, в чието поддърво са всички срещания на **x**.

Изпит по Функционално програмиране

спец. Информатика, Компютърни науки, 1 поток, и Софтуерно инженерство

14.02.2020 г.

Вариант Б

Задача 1. База данни се описва с име (низ) и размер в MB (цяло число). Сървър за данни се описва с име (низ), капацитет в MB (цяло число) и списък от бази данни. Нека е даден списък от сървъри **I**.

а) (8 т.) Да се реализира функция **mostUtilized**, която намира името на този сървър в **I**, който има най-висок коефициент на използване (отношението на сумата на размерите на базите данни в сървъра и капацитета му).

б) (12 т.) Да се реализира функция **addNew**, която добавя нов сървър с дадено име **s** и капацитет **c** в списъка и премества последователно най-големите бази данни от другите сървъри от **I** в **s** докато капацитета му позволява. Функцията да връща новия списък от сървъри след прехвърлянето.

Упътване: преместването може да се изпълни на стъпки, като на всяка стъпка се премества най-голямата база данни.

в) (Бонус 15 т.) Да се реши б) така, че капацитетът на **s** да се запълни максимално. Не е нужно решението да е оптимално по време.

Задача 2. (15 т.) Да се реализира функция **bins**, която по дадена двуместна числова функция **f** и число **z** генерира безкраен поток от всички възможни числа (евентуално с повторения), които могат да се получат чрез някакъв брой (включително и 0) последователен брой прилагания на **f** над **z**, например: **f(z, z), f(z, f(z, z)) f(f(z, z), f(f(z, z), z))**.

Пример `bins (\x y -> x^2+3*y) 2` \rightarrow [2, 10, 34, 106, 130, ...]

Задача 3. (15 т.) Разглеждаме дървета от числа с произволен брой наследници на всеки възел, с представяне по ваш избор. Числата в дървото могат да се повтарят. Да се реализира функция **maxSuccessors**, която по дадено дърво и число **x** намира най-малкото число **y**, такова, че всички срещания на **y** в дървото са в поддърво с корен **x**, или 0, ако такова число **y** няма.

Изпит по Функционално програмиране

спец. Информатика, Компютърни науки, 1 поток, и Софтуерно инженерство

14.02.2020 г.

Вариант В

Задача 1. База данни се описва с име (низ) и размер в MB (цяло число). Сървър за данни се описва с име (низ), капацитет в MB (цяло число) и списък от бази данни. Нека е даден списък от сървъри I .

а) (8 т.) Да се реализира функция **hasLargestDB**, която намира името на този сървър в I , чиято най-голяма база данни заема възможно най-малък процент от капацитета му.

б) (12 т.) Бази данни, които съвпадат по име и размер, наричаме “дубликати”. Да се реализира функция **cleanDuplicates**, която намира и изтрива всички дубликати като запазва само това копие, което е на сървъра с най-свободно място в оригиналния списък I . Функцията да връща списъка след изтриване на дубликатите, като в него не са включени сървърите, които са останали празни.

в) (Бонус 15 т.) Да се реши б) така, че максимален брой сървъри да останат празни. Не е нужно решението да е оптимално по време.

Задача 2. (15 т.) Да се реализира функция **nums**, която по дадена едноместна функция f , за която $f(x) > x$ за всяко число x , и списък от числа I , генерира безкраен поток от всички възможни числа без повторения, които могат да се получат от прилагането на f някакъв брой пъти (включително и 0) върху някое число от I .

Пример: $\text{nums } (\lambda x \rightarrow x^2+1) [1, 3, 5] \rightarrow [1, 2, 3, 5, 10, \dots]$

Задача 3. (15 т.) Разглеждаме дървета от числа с произволен брой наследници на всеки възел, с представяне по ваш избор. Числата в дървото могат да се повтарят. Да се реализира функция **minBetween**, която по дадено дърво и число x намира най-малкото число y , което е “между” две срещания на x , т.е. има срещане на y , което има x както за наследник, така и за предшественик.