

Контролно упражнение № 1 по Функционално програмиране, вариант 1

име фак. № адм. гр.

Задача 1. k -представител на естественото число $a = \overline{a_n a_{n-1} \dots a_0}$ наричаме числото $r_k(a) = \overline{a_0 a_k \dots a_{dk}}$, съставено от всички цифри, намиращи се на кратни на k позиции в a , четени от дясно на ляво. Да се дефинира функция ($ksum\ k\ a\ b$), която по зададени естествени числа k , a и b ($a \leq b$) намира сумата на всички k -представители на числата от интервала $[a, b]$. За целта да се дефинира и използва функция *accumulate*, която натрупва по определено правило стойностите на дадена функция над даден числов интервал.

Примери: $r_3(9876543210) = 0369 = 369$, $r_2(87654321) = 1357$.

Задача 2. Дадени са списък от числа $l = (c_1\ c_2\ \dots\ c_k)$ и списък от точкови двойки $change = ((a_1, b_1)\ (a_2, b_2)\ \dots\ (a_k, b_k))$. Да се дефинира функция (*replace\ l\ change*), която връща като резултат списък от вида $(d_1\ d_2\ \dots\ d_k)$, където $d_i = c_i$, ако c_i не се среща сред елементите a_s ($s = 1, 2, \dots, k$), и $d_i = b_j$, ако $c_i = a_j$.

Задача 3. Ако f и g са числови функции и n е естествено число, да се дефинира функция от по-висок ред (*switchsum\ f\ g\ n*), която връща като резултат функция, чиято стойност в дадена точка x е равна на $f(x) + g(f(x)) + f(g(f(x))) + \dots$ (сумата включва n събираеми).

Контролно упражнение № 1 по Функционално програмиране, вариант 2

име фак. № адм. гр.

Задача 1. Да се дефинира функция (*concat\ a\ b*), която връща като резултат естественото число, получено чрез “прилепване” на числата от интервала $[a, b]$ (a и b са естествени числа, $a \leq b$). За целта да се дефинира и използва функция *accumulate*, която натрупва по определено правило стойностите на дадена функция над даден числов интервал.

Примери: (*concat\ 1\ 9*) \rightarrow 123456789, (*concat\ 19\ 23*) \rightarrow 1920212223.

Задача 2. Да се дефинира функция (*mix\ a\ b*), която смесва два списъка a и b , като конструира и връща като резултат точкова двойка от списъците $(a_1\ b_2\ a_3\ b_4\ \dots)$ и $(b_1\ a_2\ b_3\ a_4\ \dots)$.

Задача 3. Да се дефинира функция (*task\ f*), която за дадена строго растяща функция f връща като резултат функция на два аргумента a и b ($a < b$), с чиято помощ се намира най-малкото число n – такова, че $f(\underbrace{f(\dots(f(a)\dots))}_n) > b$.