

Домашна работа № 1 по Функционално програмиране

специалност „Информационни системи“, I курс

2023/2024 учебна година

Решенията трябва да са готови за автоматично тестване. Важно е програмният код да бъде добре форматиран и да съдържа коментари на ключовите места. Предайте решенията на всички задачи в *един* файл с наименование *hw1_<FN>.hs*, където *<FN>* е Вашият факултетен номер.

Домашните работи се предават като изпълнение на съответното задание в курса по ФП в Moodle (<https://learn.fmi.uni-sofia.bg/course/view.php?id=10042>) най-късно до **23:55 ч. на 18.03.2024 г.** (понеделник).

Приятна работа и успех!

Задача 1

Клайд изкачва стълбище, съставено от n стъпала (n е естествено число). Клайд прави два вида стъпки - стъпка, с която изкачва 1 стъпало, и стъпка, с която изкачва 2 стъпала.

Да се дефинира функция `numStepCombinations :: Integer -> Integer`, която намира броя различни начини, по които Клайд може да изкачи стъпалата. При подаден невалиден вход да се породи грешка с подходящ текст.

Примери:

```
numStepCombinations 2    → 2
numStepCombinations 3    → 3
numStepCombinations 100  → 573147844013817084101
```

Обяснения:

Пример 1: При стълбище, съставено от 2 стъпала, Клайд може да направи две комбинации от стъпки:

1. Два пъти да изкачи по едно стъпало;
2. Един път да изкачи две стъпала.

Пример 2: При стълбище, съставено от 3 стъпала, Клайд може да направи три комбинации от стъпки:

1. 1 стъпало + 1 стъпало + 1 стъпало
2. 1 стъпало + 2 стъпала
3. 2 стъпала + 1 стъпало

Задача 2

Понятието „устойчивост на естествено число” се дефинира като броя последователни стъпки, на които трябва да се умножат ненулевите цифри на това число, на резултата от умножението и т.н., докато се получи едноцифрено число.

Например: устойчивостта на числото 273 е 2 ($2*7*3=42$; $4*2=8$);

устойчивостта на числото 277 е 4 ($2*7*7=98$; $9*8=72$; $7*2=14$; $1*4=4$);

устойчивостта на числото 392 е 3 ($3*9*2=54$, $5*4=20$; $20 \div 10=2$).

Да се дефинира функция **maxPersistenceMinSum** :: **Int** -> **Int** -> **Int**, която за даден интервал от числа [start, end] намира числото в интервала, което има максимална устойчивост, а ако има няколко такива, връща числото с минимална сума на цифрите.

Примери:

```
maxPersistenceMinSum 273 392    → 355
maxPersistenceMinSum 1000 2000 → 1679
maxPersistenceMinSum 55 105     → 77
maxPersistenceMinSum 195 756    → 679
maxPersistenceMinSum 2 85       → 77
```