

**Задачи за подготовка за първо контролно по ФП,
специалност „Информационни системи“**

Задача 1. Интересно число е естествено число, което се дели без остатък на сумата на своите цифри. Например числото 410 е интересно, тъй като $4 + 1 + 0 = 5$ е делител на 410. Напишете функция, която проверява дали дадено естествено число n е интересно.

Задача 2. Напишете функция, която връща като резултат сумата от целите числа в интервала $[a, b]$ (a и b са две дадени естествени числа, $a \leq b$), които са от вида $4k+1$ (k е цяло число) и в десетичния запис на които се съдържа цифрата 6.

Задача 3. Напишете функция, която за даден списък l , елементите на който са непразни списъци от числа, връща като резултат списък от тези елементи на l , които представляват аритметична прогресия (числова редица, в която всяко число след първото се получава, като към предишното се прибавя една и съща константа).

Задача 4. Дефинирайте функцията $\sin n x$, която приема целочисления аргумент n и реалното число x и връща n -тата частична сума на развитието в степенен ред на функцията $\sin(x)$, дефинирано като:

$$\sin(x) = \sum_{i=0}^{\infty} \frac{(-1)^i x^{2i+1}}{(2i+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

Задача 5. Нека са дадени две едноаргументни числени функции f и g и списък от числени стойности xs . Ще казваме, че функцията f доминира g върху множеството xs , ако за всяко $x \in xs$ е вярно, че $|f(x)| \geq |g(x)|$.

Дефинирайте функцията **dominates** $f g xs$, която връща резултата от проверката дали функцията f доминира g върху множеството xs .

Задача 6. Нека са дефинирани следните типове:

```
type Student = String -- име на ученик
type Subject = String -- име на предмет
type Note = Double    -- оценка

-- Запис за ученик, съдържащ име на ученик, учебен предмет и оценката на
-- ученика по дадения предмет.
type Record = (Student, Subject, Note)
```

Дефинирайте функцията **hardestSubject** :: [Record] -> Subject, която получава списък от записи за учениците от даден клас и връща името на предмета с най-ниска средна оценка за този клас.

Задача 7. Напишете на езика Haskell функция **reverseOrdSuff** :: Int -> Int, която по дадено естествено число k намира число, получено от цифрите на най-дългия строго низходящ суфикс на k , взети в обратен ред.

Примери:

```
reverseOrdSuff 37563 -> 36
reverseOrdSuff 32763 -> 367
reverseOrdSuff 32567 -> 7
reverseOrdSuff 32666 -> 6
```

Задача 8. Да се напише на Haskell функция `sumUnique :: [[Int]] -> Int`, която по списък от списъци от цели числа намира сумата на тези от числата, които са уникални в рамките на списъка, в който се срещат.

Примери:

`sumUnique [[1,2,3,2],[-4,-4],[5]] -> 9 (= 1+3+5)`

`sumUnique [[2,2,2],[3,3,3],[4,4,4]] -> 0`

`sumUnique [[1,2,3],[4,5,6],[7,8,9]] -> 45`

Задача 9. Продукт се представя с наредена двойка от вида (име, цена). Наличността в даден магазин се представя със списък от продукти.

`type Product = (String,Double)`

`type StoreAvailability = [Product]`

а) Да се напише на Haskell функция

`closestToAverage :: StoreAvailability -> String`, която намира името на продукта, чиято цена е най-близка до средната цена за всички продукти. Ако има повече от един такъв продукт, функцията да връща името на кой да е от намерените.

б) Да се напише на Haskell функция

`cheaperAlternative :: StoreAvailability -> Int`, която намира броя на продуктите, за които има продукт със същото име, но по-ниска цена.

Примери:

`store1=[("bread",1),("milk",2.5),("lamb",10),("cheese",5),("butter",2.3)]`

`closestToAverage store1 -> "cheese"`

`store2=[("bread",1),("cheese",2.5),("bread",1),("cheese",5),("butter",2.3)]`

`cheaperAlternative store2 -> 1`

Задача 10. Нека е даден списък от точки в тримерно пространство, представени като наредени тройки. Да се напише на Haskell функция

`minDistance :: [(Double,Double,Double)] -> Double`, която намира най-малкото от разстоянията между двойките точки от списъка.

Разстоянието d се дефинира по следния начин: ако разглеждаме точките $p1=(x1, y1, z1)$ и $p2=(x2, y2, z2)$, то $d(p1, p2) = (x1-x2)^2+(y1-y2)^2+(z1-z2)^2$.