

ЗАДАЧИ ЗА ЗАДЪЛЖИТЕЛНА
САМОПОДГОТОВКА
ПО
Обектно-ориентирано програмиране
Функции от високо ниво

email: kalin@fmi.uni-sofia.bg

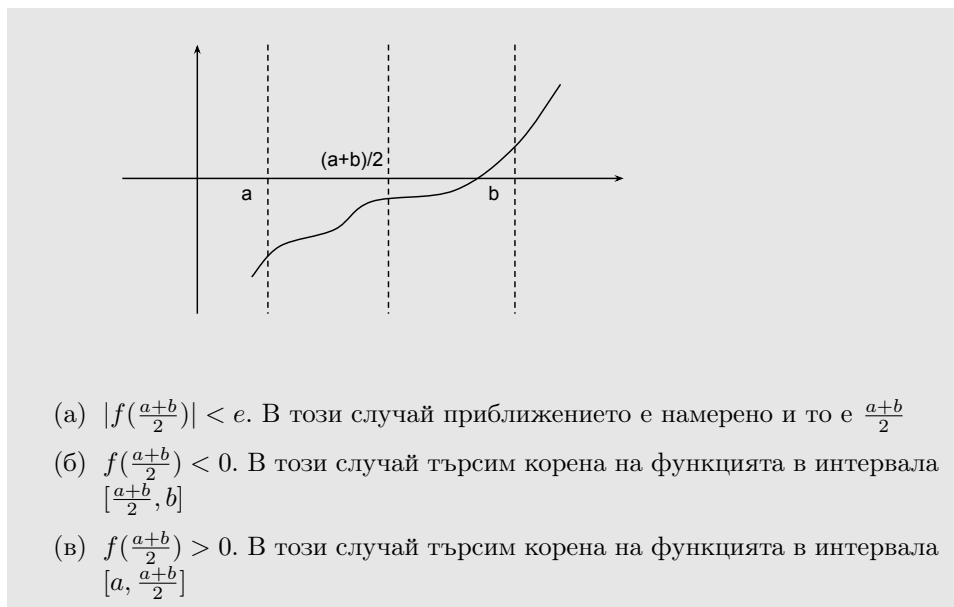
2 май 2018 г.

1. Да се дефинира функция `double root ([подходящ тип]f, double a, double b, double e)`, където $f : \text{double} \rightarrow \text{double}$ е непрекъсната и монотонна в интервала $[a, b]$ и притежава корен в него, а e е положително число. Чрез използване на двоично търсене (*bisection*), функцията `root` да намира приближение на корена на f в интервала $[a, b]$ с грешка най-много e .

Упътване:

Установете дали функцията е растяща или намаляваща. Да приемем, че функцията е растяща. За намаляващи функции алгоритъмът е аналогичен.

За всеки интервал $[a, b]$ имаме точно три възможни случая:



- Дефинирайте два варианта на функцията: итеративен и рекурсивен.
 - Тествайте функцията `root` с поне два примера.
2. Да се дефинира функция `void zip (double a1[], double a2[], double res[], int n, [подходящ тип]f)`, където `a1`, `a2` и `res` са масиви с `n` на брой елементи, а `f` е функция от тип $f : double \times double \rightarrow double$. Като резултат от работата на функцията елементите на `res` да съдържат стойностите на функцията `f` върху съответните елементи на `a1` и `a2`, така че $res[i] = f(a1[i], a2[i])$ за $i = 0..n - 1$.
- Тествайте функцията с поне два примера.
3. Функцията `zip` от предишната задача да се преобразува до шаблон, така че масивите `a1`, `a2` и `res` да са от произволен тип `T`.
- Тествайте функцията с поне два примера.
4. Функцията `zip` от предишната задача да се преобразува до шаблон, така че всеки от масивите `a1`, `a2` и `res` да са от различни помежду си типове T_1 , T_2 и T_3 , а $f : T_1 \times T_2 \rightarrow T_3$.
- Тествайте функцията с поне два примера.