**Задания 2. Функциональный стиль**

Для понимания различий между последовательным («императивным») стилем и функциональным стилем программирования напишите несколько задач в двух вариантах.

**Задача**. *Напишите программу для вычисления приближенного значения числа****е****по формуле для разложения ех в ряд Тейлора. Напишем программы на Java, написанные в традиционном императивном и функциональном стилях.*

Задача является примером задачи вычисления значения по заданной функциональной зависимости. Поскольку вычисления осуществляются приближенно, необходимо ввести дополнительный аргумент, определяющий точность вычислений. Будем считать, что задано также вещественное значение *eps*. Суммирование ряда Тейлора будет прекращено, когда очередной член ряда станет по абсолютной величине меньше *eps*.  Задача сводится к вычислению функции ех(х, eps).

Вычисление числа е в традиционном стиле

**public static double** ex(**double** x, **double** eps) {  
 **double** ex = 0;  
 **double** u = 1;  
 **int** n = 1;  
 **while** (Math.*abs*(u) >= eps) {  
 u = u \* x / n;  
 n++;  
 }  
 **return** ex;   
}

Для императивного стиля программирования характерно, что программа содержит повторяющиеся вычисления, записанные в виде цикла. Инвариантом цикла в данной программе служит тот факт, что переменная *ех* содержит частичную сумму ряда с (*n*-1) членом, переменная содержит значение очередного (следующего) члена этого ряда.

Для написания той же программы в функциональном стиле следует сначала написать функцию, определяющую зависимость *ех*от *х* и *eps*, явно. Такая зависимость может быть выражена с помощью рекуррентного соотношения, которое определяет способ вычисления частичных сумм ряда. Если уже имеется вычисленная частичная сумма ряда *Sn* и очередной член ряда *Um* то вычисления можно заканчивать, если значение *Un* меньше заданного *eps*. Если Нет, то требуется добавить очередной член ряда к сумме, вычислить значение очередного члена ряда и вызвать ту же функцию рекурсивно.

1. Вычисление числа *е* в функциональном стиле

**public static double** exRec(  
 **double** x, **double** eps, **double** sum, **double** u, **int** n) {  
 **return** Math.*abs*(u) < eps ? sum :  
 *exRec* (x, eps, sum + u, u \* x / n, n + 1) ;  
}

Для функционального стиля программирования характерно, что программа представляет собой набор функций, каждая из которых является суперпозицией других функций и операций, возможно, с прямым или косвенным рекурсивным обращением к самой определяемой функции. В данном примере основную работу выполняет рекурсивная функция exRec.

Задачи для самостоятельного решения. ***Решите 2 задачи на выбор.***

В заданиях требуется написать программу с использованием двух различных стилей программирования — традиционного императивного стиля и функционального стиля. Можно использовать любой из традиционных языков программирования (Python, С#, C++, Java …..).

2.1. Напишите функцию для нахождения заданного значения в упорядоченном массиве целых чисел методом двоичного поиска.

2.2. Напишите программу для нахождения площади заданного выпуклого многоугольника, если заданы координаты всех его вершин. Можно считать, что вершины упорядочены таким образом, что каждая следующая вершина связана с предыдущей ребром многоугольника.

2.3. Напишите программу для вычисления n-то члена числовой последовательности, заданной следующим рекуррентным соотношением: ао = ai= 1, а n+2 =3a n - 2a n-1 + 1 при n > 1.

2.4. Напишите программу для вычисления приближенного значения функции cos х при заданном значении х. Для вычислений используйте разложение косинуса в ряд Тейлора в окрестности нуля. Примените периодичность тригонометрических функций для приведения аргумента к окрестности нуля для повышения точности вычислений.

2.5. Напишите программу для нахождения минимального из чисел, являющихся максимальными в каждой из строк заданной прямоугольной матрицы.

2.6. Напишите программу для нахождения длины максимальной возрастающей подпоследовательности рядом стоящих элементов в заданном массиве.

2.7 Близнецами называется пара натуральных чисел, каждое из которых равно сумме делителей другого числа. Например, числа 220 и 284 — близнецы (проверьте!). Напишите программу для нахождения первых п пар близнецов

2.8. Совершенным числом называется натуральное число, равное сумме всех своих делителей, включая единицу, но исключая само это число. Например, число 28 — совершенное, поскольку 28 = 1+ 2 + 4 + 7 + + 14. Напишите функцию для нахождения первых п совершенных чисел.