

Авторска справка на гл. ас. д-р Бойко Банчев за научните приноси на трудовете, представени за участие в конкурса за доцент по информатика (езици на информатиката), обявен от Института по математика и информатика в ДВ, бр. 2 от 8.1.2016 г.

(В квадратни скоби са посочени номерата на публикациите според приложения отделно списък.)

В [1] се привлича внимание върху непълната адекватност на традиционните итеративни конструкции в езиците за програмиране при изразяване на редица алгоритмични схеми. За преодоляване на този недостатък се предлага алтернативно, по-общо множество от итеративни конструкции – консумативни команди. Посочва се, че последните лежат в основата на нов, консумативен модел на пресмятане, разширяващ обичайния императивен модел от гледна точка на изразяването на повторност.

В [3] са предложени няколко конструкции за усъвършенстване на изразите в императивни езици. С тяхна помощ става възможно да се съвместяват каквито и да е операции с присвояване и с условно присвояване, във варианти с преди- и с последствие. Обособява се клас от операции, наречени съюзи; предлагат се и конкретни съюзи, произтичащи от консумативния модел на пресмятане [1]. Семейство конструкции – енторки – позволяват агрегиране на действия, на техните аргументи и резултати.

Работите [2,8,16] са посветени на изследване на възможностите за оформяне на общ подход, преди всичко обща концептуална основа, за третиране на понятието структура и на структурността в програми и по-общо в информатични обекти.

Дисертацията [18] разглежда тематиката на [1] и [3] и редица други програмни конструкции от гледна точка на определеното в нея понятие явност и някои близки до него свойства на програми. Част от нея е посветена на общия подход към структурността от [2,8,16]. Формулирани са принципи на такава структурност и са посочени примерни понятия, с които тя се описва.

[18] съдържа и резултати, които засега не са разглеждани в отделни публикации:

- направена е обща сравнителна съпоставка на езиците за програмиране като изразно средство с говоримите (преди всичко естествените) езици и този на математиката;
- формулирани и съотнесени са ред фактори за възприемане на програмен текст;
- разгледана е една типична, обща употреба на метаезика BNF и на тази основа е предложена конструкция за усъвършенстването му.

В публикацията [6] се разглеждат проблемите с определяне на операции за намиране на целочислено частно и остатък при делене и близки до тях в езиците за програмиране. Предлага се нов подход, чрез който се постигат едновременно по-голяма общност, последователност и съгласуваност при определяне на такива операции.

В [4] е предложен език за създаване на планиметрични чертежи. Той позволява описване на планиметрични построения и нанасяне в чертежа на разнообразни ви-

дове обозначения. Оригинални езикови решения позволяват йерархично задаване на геометрични фигури и избирателно визуализиране и именуване на различните части на дадена конструкция.

На основата на анализ на потребностите за описване на свойства на програми и съпоставяне с традиционния математически запис и със съществуващи езици за аотиране, в [5] е предложен език за аотиране на програми. Езикът съдържа средства за изразяване на аритметични и логически зависимости, за образуване и уточняване на съдържанието на множества и редици, за почленно преобразуване и др., приложим е към програми на който и да е език за програмиране и сам допуска формално, вкл. програмно третиране.

Работата [11] е описание и критичен обзор на функционалния език за програмиране Рефал, вкл. в контекста на функционалното програмиране изобщо и в историко-концептуален план.

Статията [7] описва обширен програмноезиков експеримент: решаване на една и съща задача по различни начини с помощта на десетки езици за програмиране. Приведени са наблюдения и изводи относно практическата използваемост на езиците и приложимостта на еталонни задачи за сравнително съпоставяне на езици.

[10], [13] и [14] са посветени на целите, съдържанието и средствата за преподаване на информатика и програмиране в училище, включително различни езикови аспекти. Обосновават се няколко тези, например следните. Съвременен динамичен език за програмиране е в много по-голяма степен от традиционните Basic, Pascal, C/C++, Java и др. подходящ за преподаване на информатика. Подходящо уточнена разновидност на функционалния стил на програмиране е особено сполучлив вариант за плавно, в хармония с обучението по математика навлизане в информатиката. За преподаване на информатични концепции могат да се използват не само езици за програмиране, но и широк кръг средства с квазипрограмни възможности.

Статиите [9] и [12] описват проектиран от автора функционален език U за учебно и експериментално програмиране. Езикът съдържа редица оригинални черти, които предполагат удобна приложимост за решаване на широк кръг задачи, наред с много голяма простота на конструкциите на самия език. В [15] се разглежда тъй наречената функционална геометрия – микроезик за образуване на фигури чрез прилагане на композиращи операции над по-прости фигури – и по-специално усъвършенстването на модела на функционалната геометрия в езика U.

В [17] се разглеждат различни от гледна точка на свойствата им в езиков план решения на една традиционна като упражнение по програмиране задача.

Подпис:

(Бойко Банчев)