

СОПОСТАВЛЕНИЕ С ОБРАЗЦОМ В PASCALABC.NET

Михалкович С.С.

*Южный федеральный университет,
факультет математики, механики и компьютерных наук
E-mail: miks@math.sfedu.ru*

Каждый год язык программирования PascalABC.NET совершенствуется – в нём появляются новые конструкции, имеющиеся в современных языках программирования [1–4]. Данные конструкции реализуются как синтаксический сахар, методы реализации которого рассмотрены в [5].

Одним из последних нововведений в PascalABC.NET является конструкция сопоставления с образцом (Pattern Matching). Данная конструкция пришла из функциональных языков и сейчас активно внедряется в такие универсальные языки как C#, Java и другие.

Смысл сопоставления с образцом коротко поясняется как «case по типам»: действия выполняются в зависимости от типа объекта. Для обучения конструкция Pattern Matching обычно используется в теме «Объектно-ориентированное программирование» рядом с такими темами как «Полиморфизм», «Операции is и as» и «Полиморфные контейнеры».

Рассмотрим несколько примеров, поясняющих синтаксис сопоставления с образцом в PascalABC.NET.

Пример 1. Увеличение радиуса всех кругов в полиморфном контейнере.

```
var l := new List<Object>;  
...  
foreach var x in l do  
    if x is Circle(var c) then  
        c.Radius += 1;
```

Здесь в полиморфном контейнере графических объектов проверяется, что x – это круг, и если это так, то в переменную c записывается ссылка на круг, после чего через эту переменную мы увеличиваем радиус круга.

Пример 2. Вычисление площадей всех фигур в полиморфном контейнере.

```
var l := new List<Object>;  
l.Add(new Line);  
l.Add(new Circle(10,10,5));  
l.Add(new Rectangle(10,10,20,10));  
foreach var x in l do  
    match x with  
    Line(var ll): Println(0);  
    Circle(var c): Println(c.Radius*c.Radius*Pi);
```

```

    Rectangle(var r): Println(r.Width*r.Height);
end;

```

В данном примере в полиморфном контейнере фигур для каждой фигуры вычисляется площадь. Это является альтернативой традиционному способу – созданию виртуального метода вычисления площади.

Пример 3. Вычисление значения выражения, заданного деревом разбора.

```

function Eval(e: Expr): real;
begin
    match e with
        Cons(c): Result := c;
        Neg(n): Result := -Eval(n);
        Add(l,r): Result := Eval(l) + Eval(r);
        Mult(l,r): Result := Eval(l) * Eval(r);
    end;
end;

```

Данный пример показывает, сколь компактно и интуитивно понятно может вычисляться значение выражения, заданного деревом разбора: в зависимости от типа узла вычисления производятся по различным формулам.

Пример 4. Упрощение выражений (глубокий Pattern matching).

```

function Simplify(e: Expr): Expr;
begin
    match e with
        Mult(Cons(c),Cons(c1)): Result := ConsC(c*c1);
        Add(Cons(c),Cons(c1)): Result := ConsC(c+c1);
        Mult(Cons(c),ex) when c=1:
            Result := Simplify(ex);
        Mult(Cons(c),ex) when c=0: Result := ConsC(0);
        Add(Cons(c),ex) when c=0:
            Result := Simplify(ex);
        Add(Add(ex,Cons(c)),Cons(c1)):
            Result := AddC(Simplify(ex),ConsC(c+c1));
        Mult(Mult(ex,Cons(c)),Cons(c1)):
            Result := MultC(Simplify(ex),ConsC(c*c1));
        else Result := e;
    end;
end;

```

В данном примере демонстрируется рекурсивная функция Simplify, которая упрощает дерево разбора арифметического выражения, используя условия when на параметры в образцах, а также «глубокое» сопоставление с образцом (в данном случае в качестве образца выступают поддеревья глубины 2 или 3).

Использование конструкции match сопоставления с образцом позволило оосовременить и упростить курс «Основы программирования» [6] для студентов направления «Фундаментальная информатика и информационные технологии» мехмата ЮФУ.

Литература

1. *Бондарев И.В., Белякова Ю.В., Михалкович С.С.* Система программирования PascalABC.NET: 10 лет развития / XX Научная конференция «Современные информационные технологии: тенденции и перспективы развития». Материалы конференции. Ростов-на-Дону, 2013. С. 69–71.
2. *Бондарев И.В., Михалкович С.С.* Система программирования PascalABC.NET: новые возможности 2015-16 гг. / Труды XXIII Научно-методической конференции «Современные информационные технологии: тенденции и перспективы развития». Ростов-на-Дону: Изд-во ЮФУ, 2016. С. 69–71.
3. *Михалкович С.С.* Оператор yield в языке PascalABC.NET и его использование в курсе «Основы программирования» / Современные информационные технологии: тенденции и перспективы развития Материалы XXIV научной конференции. 2017. С. 130-132.
4. *Бондарев И.В., Михалкович С.С.* Система программирования PascalABC.NET: 15 лет развития / Труды XXV Научно-методической конференции «Современные информационные технологии: тенденции и перспективы развития». Ростов-на-Дону: Изд-во ЮФУ, 2018. С. 31–34.
5. *Михалкович С.С.* Методы реализации синтаксически сахарных расширений в компиляторах. Известия высших учебных заведений. Северо-Кавказский регион. Серия: Технические науки. 2017. № 3 (195). С. 49-55.
6. *Михалкович С.С.* Курс «Основы программирования» на базе системы PascalABC.NET / Научная конференция «Современные информационные технологии и ИТ-образование». Материалы конференции. Москва, 2009. № 5, С. 385.