Udaff — русский пиктографический Python. От элементарных алгоритмов до гомоморфного шифрования

Стас Фомин*, stas-fomin@yandex.ru

2020
Январь

Аннотация

Автор часто встречал изобретения «специальных языков для обучения программированию», кривых и убогих, с жалкой инфраструктурой и поддержкой, отрезанных от мирового мейнстрима программирования и обучения, при этом навязываемых в школах и ВУЗах. Иногда необходимость таковых обосновывается требованиями «русскоязычности» ключевых слов.

Предлагается шире использовать Python — язык, годный от обучения дошкольников до глубокого профессионального использования во многих областях. А если кому-то нужна именно русскоязычность ключевых слов — туда ее можно добавить, что и проделал автор, не нарушив совместимости. Кроме русскоязычности добавлены и пиктограммы, эмоджи — возможно это тоже поможет в обучении начинающим, а возможно это пригодится и взрослым — для компактизации описаний нетривиальных алгоритмов.

1 Проблема

В русскоязычном пространстве наблюдается странный эффект изобретения «русскоязычных» языков программирования, и попыток навязать их обучающимся в Школе и ВУЗе.

См. например Кумир [1], SLang [2] (не путать с еще одной самоделкой — СЛанг [3])... множество их.

*При поддержке гранта РФФИ 19-01-00702А

С одной стороны, вроде как основания есть — из-за специфики высшего образования в РФ (бесплатное образование, оплачивается ВУЗам МинОбром подушевым образом, отчислять невыгодно, и даже запрещено [4]), в ВУЗы на околопрограммиские специальности попадает множество немотивированных и функционально необразованных кадров, не способных понимать даже текст с десятком ключевых слов на английском. С другой стороны, возможно успех таких платформ типа 1С именно этим и обусловлен, да и есть немало энтузиастов, которые считают, что русификация ЯП — полезна [5], и учить на русифицированных языках эффективно [6].

Однако проблема всех этих самоделок в том, что язык — это не только синтаксис грамматики в BNF на полстраницы, а это инфраструктура:

- Редакторы, IDE, поддерживающие 100500 удобных фич, не говоря уже о обязательной «построчной» отладке.
  - Разработчики «Русских ЯП» пытаются делать некие подобия IDE, благо сейчас это можно слепить из каких-нибудь готовых компонентов (модуль текстового редактора с подсветкой, MDI интерфейс с менюшками) — но все что получается, это скажем прямо, уровень 90х, и сравнивая это с бесплатным и open-source «швейцарским ножем» Visual Studio Code, (не говоря уже о коммерческих IDE) хочеться только плакать от жалости.
- Инфраструктура пакетов (сами библиотеки, пакетные менеджеры), 100500 пакетов для решения всего скучного и типового, не говоря уже о нетривиальных платформах (быстрый вход в разработку игр, математические методы и AI и т.п.)
- Правильные концепции и парадигмы языка, проверенные десятилетиями обкатки на миллионном комьюнити профессионалов.
- Возможность получения профессиональных НАВЫКОВ, вшитых на уровень костного мозга, которых можно применить в профессиональной разработке для решения реальных проектов. Чтобы стартовав с элементарных алгоритмов и простых поделий, можно было эволюционировать в профессионала.

Но если «показать что-то про программирование на русском», на уровне «операторы-ветвление-цикл-функция-рекурсия» для совершенно левых людей (которым максимум в 1С в жизни придется что-то подправить) — допустимо, то совращать «девственно» чистых школьников кривыми поделиями, отрубая им прямой выход к реальной разработке и отбивая желание программировать («пробовал ваше программирование на К... » — ничего не работает, тормозит, криво, неудобно, долбайтесь сами), как минимум неэтично, хотя к сожалению, уголовно ненаказуемо.

Кстати, иногда изобретают «еще одну платформу для обучения» даже не для русификации языка, а для, скажем так, «реанимации стюардессы», типа паскаля, но проблемы остаются те же — какое-то свое подмножество языка, унылые среды разработки, все сбоку от майнстрима и сообщества [7].

Возникает дилемма — как бы убить двух зайцев — пользуясь одной платформой дать возможность неодаренным и негодным студентам и школьникам [8] «попробовать программирование», а продвинутым, тут же, в том же классе, на том же софте, дать возможность уйти в отрыв и стать настоящими профессионалами.

2 Решение

Итак, уже есть Python, идеальный язык для обучения, рожденный как язык для обучения, десятилетиями использующийся от везде — от уровня младших школьников, до профессионалов, как в программировании, так и в куче научных областей — будь то астрономия, биоинформатика, статистика... везде.

Знание его полезно, если не сказать необходимо, даже тем, кто не программист, если деятельность хоть как-то интеллектуальна (да, даже если экономист-юрист уровня выше чем «за рубль покупаем, за три продаем, на эти два процента и живем»).

К нему есть куча IDE, включая прекрасную поддержку даже в бесплатном и свободном Visual Studio Code (не нужно изобретать страшные велосипеды [9]), есть 100500 пакетов, платформы для написания всего — игр, десктоп и вебприложений, даже мобильной и IOT разработки... Впрочем, все это банально и очевидно.

Наша идея [10] — добавить еще «маленькую лесенку» снизу, разрешив писать ключевые слова и базовые функции на русском, для тех, кто вот только стартует, не потеряв никаких возможностей Python.

Для этого можно воспользоваться идеей «пользовательских кодировок», перекодирующих файл с программным кодом при открытии. В результате, несколько десятков ключевых слов Python приводятся к каноническому английскому виду, их понимает и язык, и отладчики и т.п. А обучающийся имеет полную возможность, по мере изучения, постепенно заменить «русские аналоги» их оригинальными ключевыми словами, и с улыбкой (но без ненависти) забыть свои самые первые программы.

Английский	Русский
and	и

Английский	Русский
as	как
assert	проверить
break	прервать
class	класс
continue	продолжить
def	функция
del	удалить
elif	ежели
else	иначе
except	случись
exec	выполни
finally	наконец
for	перебор
from	из
global	глобальное
if	если
import	подключить
in	в
is	суть
lambda	лямбда
not	не
or	или
pass	ничего
print	печать
raise	паника
return	вернуть
try	пробовать
while	повторять
with	пусть
yield	вернуть
range	интервал

Лично я (Стас Фомин), не считаю, что это необходимо. На мой взгляд лучше таки выучить эти пару десятков ключевых слов Python на английском, и отсеять «электорат» с уровнем IQ меньше веса.
Но идя путем «выбора меньшего зла» (*toss a coin to...*), я уверен, что этот подход лучше изобретения кривых велосипедов от скучающих преподавателей, которые будут калечить поколения школьников и студентов. Т.е. я был бы рад, если все это не пригодилось при обучении, но при встрече с

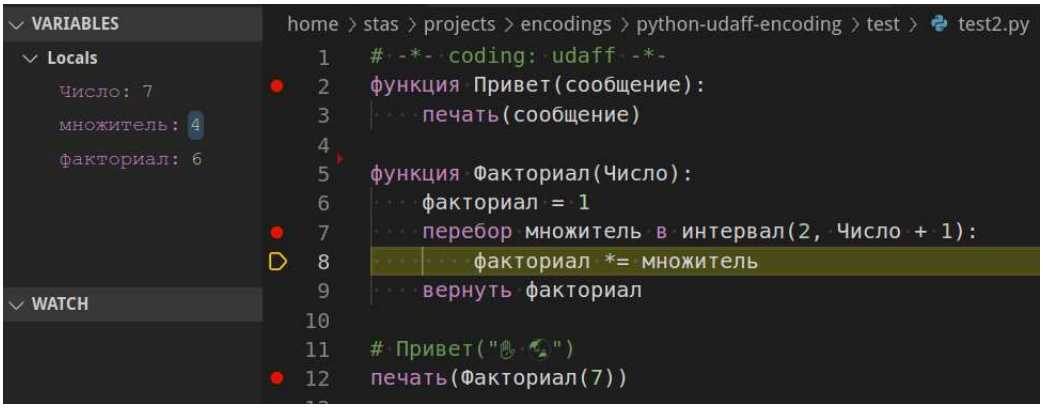

Рис. 1: Пример отладки «факториала

изобретателем очередного «национального языка», у меня будет куда его конструктивно послать.

С другой стороны, может это пригодится где-то, где Python используется как DSL для какой-нибудь бизнес-логики — короткие функции, которых должны править и вычитывать специалисты в предметной области, может это пригодится и там.

Более того, можно привлечь к программированию, совсем, так сказать, правополушарных людей, и использовать вместо идентификаторов функций и переменных значки emoji.

Если серьезно, то можно использовать и имеющуюся в юникоде «математику», и символы-эмотиконы, для того, чтобы компактифицировать реальные работающие алгоритмы, для преподавания или верификации, особенно нетривиальных алгоритмов (автор пробовал описывать блокчейны и гомоморфное шифрование).

Ну а название «udaff», отсылает к популярному времен начала Рунета ресурсу, который прекрасно иллюстрировал идею, что буквоедство и грамотность не так важны, как суть и контент, а иногда такое «грязное языковое хакерство», как кстати, в предложенном решении, даже весело. Кстати, тут несложно сделать и поддержку разных версий «падонкоффского языга».

Список литературы

[1] Доклады о языке Кумир, `http://0x1.tv/Kumir`

[2] «Обучающая среда по программированию на базе СПО (Валерий Лаптев, OSEDUCONF-2019)», `http://0x1.tv/20190127E`

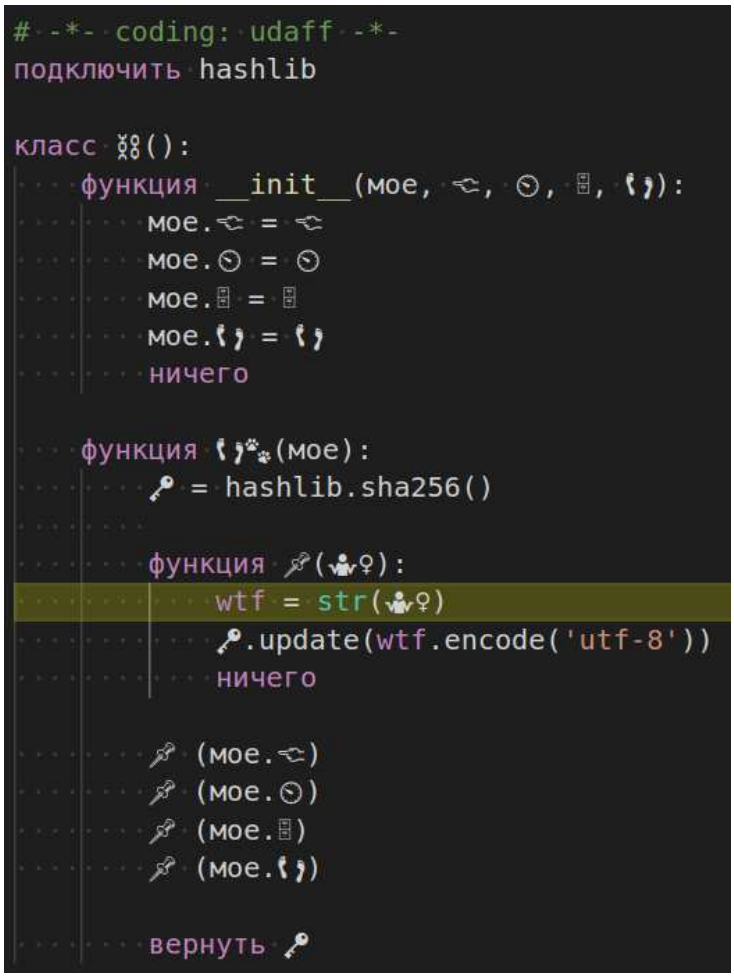

Рис. 2: Часть работающего «русифицированного» блокчейна

[3] «Проект СЛанг — текущее состояние и перспективы (Алексей Канатов, SECR-2017)», http://0x1.tv/20171021CA

[4] «Отчислять отстающих запрещено», https://youtu.be/LDdgdKI20cU?t=1351

[5] «Русификация языка – это отдельная большая тема, и её польза лично мне очевидна», http://www.0x1.tv/20191205AD#comment-4763683278

[6] «учить на русифицированных языках эффективно», https://youtu.be/LDdgdKI20cU?t=1503

[7] «Несколько причин забыть PascalABC.Net», https://habr.com/ru/post/417229/

[8] «Ваши студенты олигофрены?», https://youtu.be/LDdgdKI20cU?t=1144

[9] «IDE для изучения Python (Николай Попов, OSEDUCONF-2015)», http://0x1.tv/20150124G

[10] «Udaff — русификация Питона», https://github.com/belonesox/python-udaff-encodi