

## **Tomasz Bogdal**

2012-Present Arduino-CMake - Open Source Project

Creator/Maintainer  $\Theta$ 

2009-Present [NComputing] Linux Software Developer **Build Tools Specialist, Automation,** Embeded Firmware (ARM). Imprivata. Virtualization/Cloud. HDX N-Series

2009-2010 [CocoLab] Product Release and Management Consulting

2009 [Cyfronet AGH] Software Developer [Internship] Monitoring and Metering software for PLGrid infrastructure (Java/GWT)

2009 Developed teaching aids for Numerical Methods and Simulation class

WSZiB University - PyMOIS

https://pypi.python.org/pypi/PyMois/0.1

2006 - 2009 WSZiB University [Polish] Wyższa Szkoła Zarządzania i Bankowości w Krakowie - Wydział Informatyki

2007 - 2008 Computer Science tutoring and freelance Java programming

2004-2006 Computer Science Tutoring Programming, networking

## **EXPERIENCE**

2002-2006 High School [Polish] Liceum Ogólnokształcace

Ojców Pijarów Krakow, Poland

1999-2001 Intermediate High School [English]

Shorewood, Milwaukee, USA

1991-1999 Elementary School [French] Montreal, Quebec, Canada

**EDUCATION** 

1986

Initail commit: Born, Krakow, Poland







Electronics Interests Reverse Engineering Rock Climbing Hardware Hacking

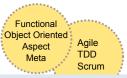






Windows 95 - Win 7 Linux Gentoo, Arch, Ubuntu, SLED, Centos Mac OS X 10 2 - 10 6 Solaris

OS



```
# head -n 16 languages known.py
languages = [
    ('Pvthon'.
                   features('dynamic', 'prototype', 'powerful')),
    ('C',
                   features('system', 'embeded', 'performance')),
                   features('speed', 'object-oriented', 'system')),
    ('C++',
    ('Bash',
                   features('system', 'commands', 'shell')),
    ('SH',
                   features('system', 'commands', 'shell', 'legacy')),
    ('JavaScript', features('browser', 'client', 'web')),
                   features('os-independent', 'web', 'enterprise')),
    ('Java',
    ('Pascal',
                   features('learning', 'readability')),
                   features('web', 'server', 'web-app')),
    ('PHP',
    ('Assembler', features('bare-metal', 'ia-32', 'performance')),
for language, features in languages:
    if problem.requirements in features:
        print 'I can solve the problem with:', language
        break
else:
    print 'No language available, I need to learn something new.'
```