Cultural biases in programming languages

Pierre Depaz – NYU Berlin

Introduction

An arbitrary tendency towards preferring one thing [over another], for reasons that can stem from one’s culture.

Gay, P. du, Hall, S., Janes, L., Madsen, A. K., Mackay, H., & Negus, K. (2013). Doing Cultural Studies: The Story of the Sony Walkman. SAGE.

Introduction

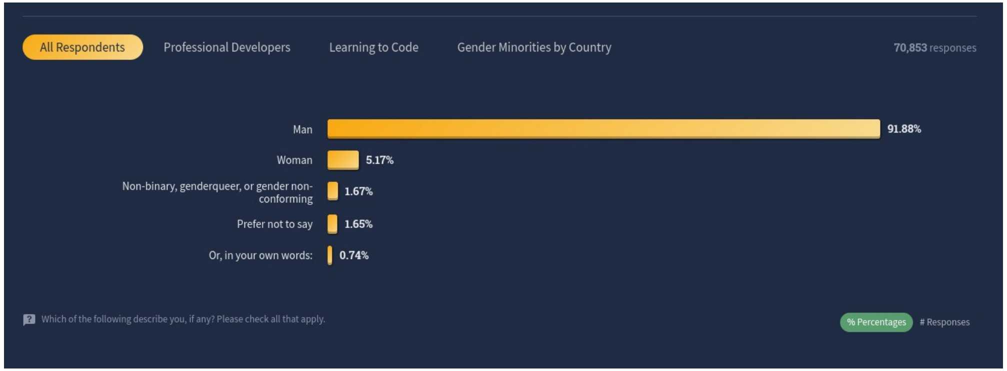
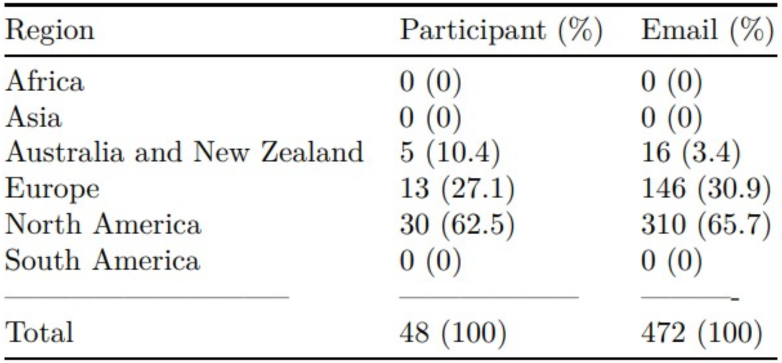


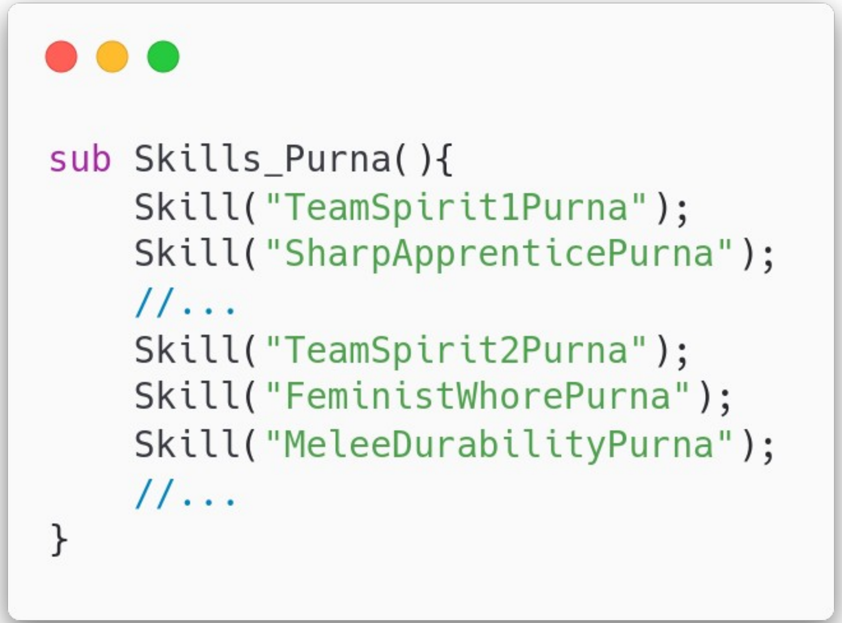
Table 1: Regional Diversity



Region	Participant (%)	Email (%)
Africa	0 (0)	0 (0)
Asia	0 (0)	0 (0)
Australia and New Zealand	5 (10.4)	16 (3.4)
Europe	13 (27.1)	146 (30.9)
North America	30 (62.5)	310 (65.7)
South America	0 (0)	0 (0)
Total	48 (100)	472 (100)

StackOverflow Developer Survey, 2022
Gupta, H. (2016). (Lack Of) Representation of Non Western World in process of creation of Web standards. arXiv:1609.01996 [Cs]. http://arxiv.org/abs/1609.01996

Introduction



```
sub Skills_Purna(){
    Skill("TeamSpirit1Purna");
    Skill("SharpApprenticePurna");
    //...
    Skill("TeamSpirit2Purna");
    Skill("FeministWhorePurna");
    Skill("MeleeDurabilityPurna");
    //...
}
```

Data0.pak\Data\default_player_setup.scr

Techland, Dead Island 2, 2011, Poland.

Introduction

English-American bias embodied in encodings and denominations

e.g. ASCII, *master/slave*, *blacklist/whitelist*

Introduction

Are technical artefacts orthogonal to cultural concerns?

Or are they manifest representations of the “right” way of doing things, of seeing the world?

SOUTHERN STATE PARKWAY. LONG ISLAND. N. Y.
S. P. 10

Southern State Parkway, Long Island, N. Y, 1930, Boston Public Library

Winner, L. (1980). Do Artifacts Have Politics? Daedalus, 109(1), 121–136.

Introduction

If computers are really the tools we use to write, to design, to play with ideas and shapes and images, they should not be addressed with the language of desktop calculators. Moving out of the impasse also would require the reconstruction of our cultural assumptions about hard logic as the "law" of thought.

Turkle, S., & Papert, S. (1992). Epistemological Pluralism and the Revaluation of the Concrete. The Journal of Mathematical Behavior.

Introduction

How can we assess whether a programming language’s design is culturally biased?

And in which specific ways?

Introduction

1. Values in programming languages
2. Contrasting estoreric languages
3. Cultural and technical biases

Values in programming languages

A programming language is a technical artefact, with ***structural*** and ***functional*** properties.

Turner, R. (2014). Programming Languages as Technical Artifacts. Philosophy & Technology, 27(3), 377–397.

Kroes, P.. (2010). Engineering and the dual nature of technical artefacts. Cambridge Journal of Economics, 34(1), 51–62.

Values in programming languages

Functionally, programming languages should enable the *ease* of creation of *correct*, *fast* and *productive* software.

Values in programming languages

Structurally, programming languages should have the desired properties:

- Clarity/readability
- Correctness
- Expressiveness/writeability

Stansifer, R. (1994). Study of Programming Languages, The (1st edition). Prentice Hall.

Sebesta, R. W. (2018). Concepts of Programming Languages (12th edition). Pearson.

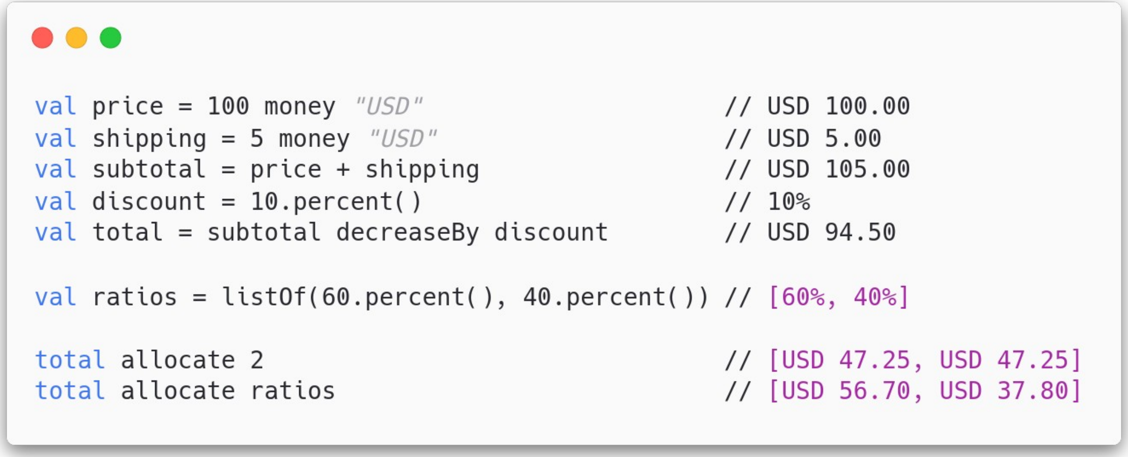
Values in programming languages

Programming languages are also the result of preferences for different ways of doing things.

No classes
No inheritance
No garbage collection
The Rust Programming Language
Anarcho-communism
imgflip.com

Values in programming languages

Primitives and ***first-class citizens*** are preferred ways of expressing a problem.

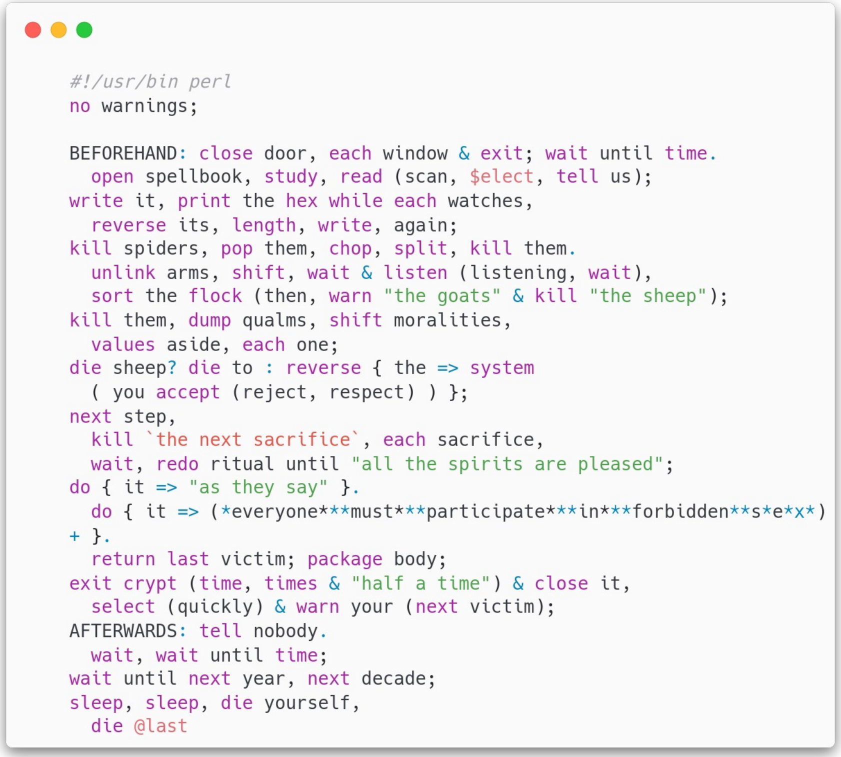


```
val price = 100 money "USD"                     // USD 100.00
val shipping = 5 money "USD"                    // USD 5.00
val subtotal = price + shipping                 // USD 105.00
val discount = 10.percent()                     // 10%
val total = subtotal decreaseBy discount        // USD 94.50

val ratios = listOf(60.percent(), 40.percent()) // [60%, 40%]

total allocate 2                                // [USD 47.25, USD 47.25]
total allocate ratios                           // [USD 56.70, USD 37.80]
```

R. J. Popplestone: The Design Philosophy of POP-2. in: D. Michie: Machine Intelligence 3, Edinburgh at the University Press, 1968.

Costa, E. (2024). Eriksencosta/money [Kotlin]. https://github.com/eriksencosta/money

Counter-styles of programming



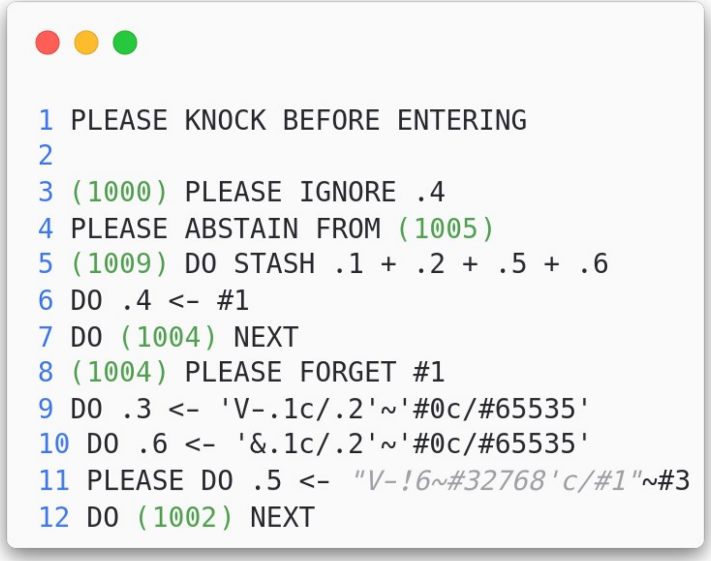
```
#!/usr/bin perl
no warnings;

BEFOREHAND: close door, each window & exit; wait until time.
  open spellbook, study, read (scan, $elect, tell us);
write it, print the hex while each watches,
  reverse its, length, write, again;
kill spiders, pop them, chop, split, kill them.
  unlink arms, shift, wait & listen (listening, wait),
  sort the flock (then, warn "the goats" & kill "the sheep");
kill them, dump qualms, shift moralities,
  values aside, each one;
die sheep? die to : reverse { the => system
  ( you accept (reject, respect) ) };
next step,
  kill `the next sacrifice`, each sacrifice,
  wait, redo ritual until "all the spirits are pleased";
do { it => "as they say" }.
  do { it => (*everyone***must***participate***in***forbidden**s*e*x*)
+ }.
  return last victim; package body;
exit crypt (time, times & "half a time") & close it,
  select (quickly) & warn your (next victim);
AFTERWARDS: tell nobody.
  wait, wait until time;
wait until next year, next decade;
sleep, sleep, die yourself,
  die @last
```

kck. (2017). Black Perl updated for Perl 5. PerlMonks. https://www.perlmonks.org/?node_id=237465

Esolangs

“An esolang (a portmanteau of “esoteric” and “language”) is not designed to solve practical issues, even esoteric ones. Instead, esolangs ask the programmer to puzzle through counterintuitive rules that challenge them or invite reflection on the act of programming.”

– Daniel Temkin

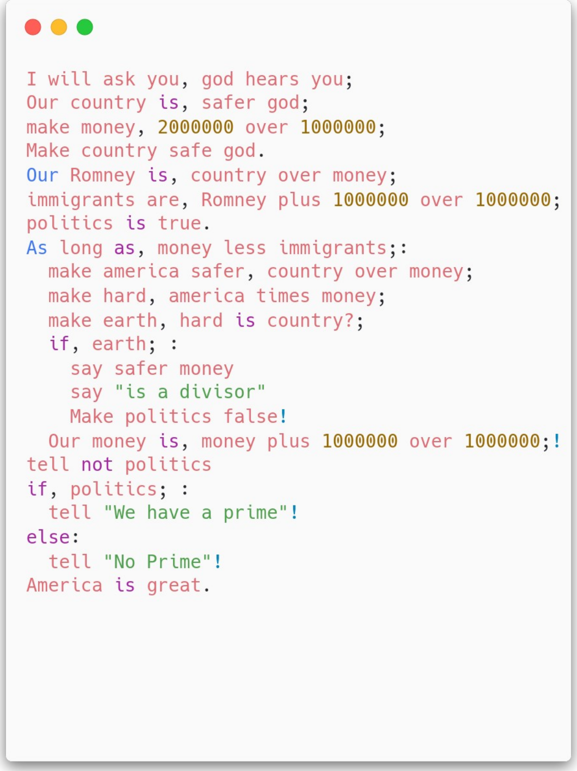


```
PLEASE KNOCK BEFORE ENTERING

(1000) PLEASE IGNORE .4
PLEASE ABSTAIN FROM (1005)
(1009) DO STASH .1 + .2 + .5 + .6
DO .4 <- #1
DO (1004) NEXT
(1004) PLEASE FORGET #1
DO .3 <- 'V-.1c/.2'~'#0c/#65535'
DO .6 <- '&.1c/.2'~'#0c/#65535'
PLEASE DO .5 <- "V-!6~#32768'c/#1"~#3
DO (1002) NEXT
```

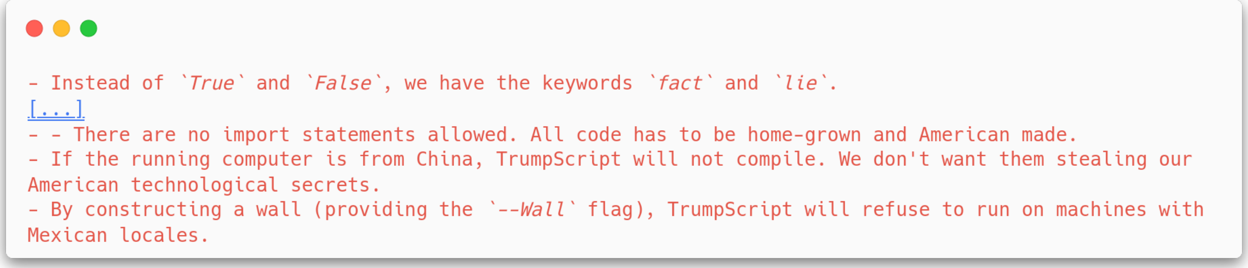
Temkin, D. (2023). The Less Humble Programmer. Digital Humanities Quarterly, 17(2). https://www.digitalhumanities.org/dhq/vol/17/2/000698/000698.html

Intercal, Donald R. Woods & James M. Lyon 1973

Esolangs – TrumpScript



```
I will ask you, god hears you;
Our country is, safer god;
make money, 2000000 over 1000000;
Make country safe god.
Our Romney is, country over money;
immigrants are, Romney plus 1000000 over 1000000;
politics is true.
As long as, money less immigrants;:
  make america safer, country over money;
  make hard, america times money;
  make earth, hard is country?;
  if, earth; :
    say safer money
    say "is a divisor"
    Make politics false!
  Our money is, money plus 1000000 over 1000000;!
tell not politics
if, politics; :
  tell "We have a prime"!
else:
  tell "No Prime"!
America is great.
```



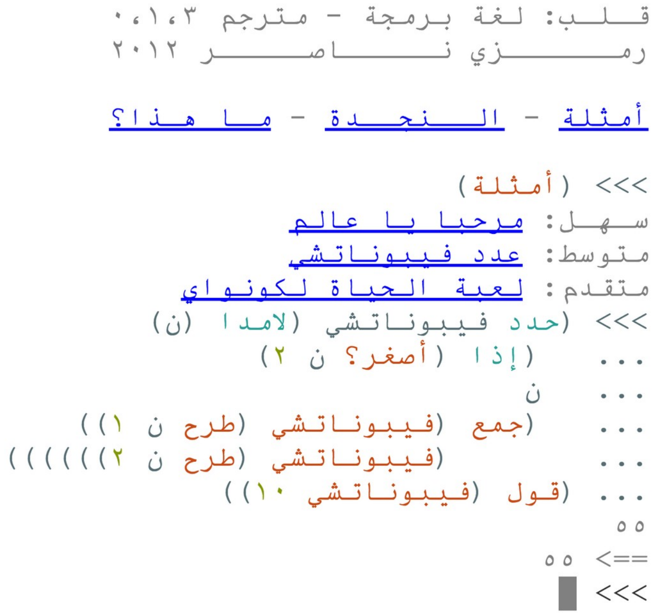
```
- Instead of `True` and `False`, we have the keywords `fact` and `lie`.
[...]
- - There are no import statements allowed. All code has to be home-grown and American made.
- If the running computer is from China, TrumpScript will not compile. We don't want them stealing our
American technological secrets.
- By constructing a wall (providing the `--Wall` flag), TrumpScript will refuse to run on machines with
Mexican locales.
```

Shadwell, S. (2016). Samshadwell/TrumpScript [Python]. https://github.com/samshadwell/TrumpScript

Esolangs - قلب

“All of the tools that I use, start to fall apart when they have to handle قلب”

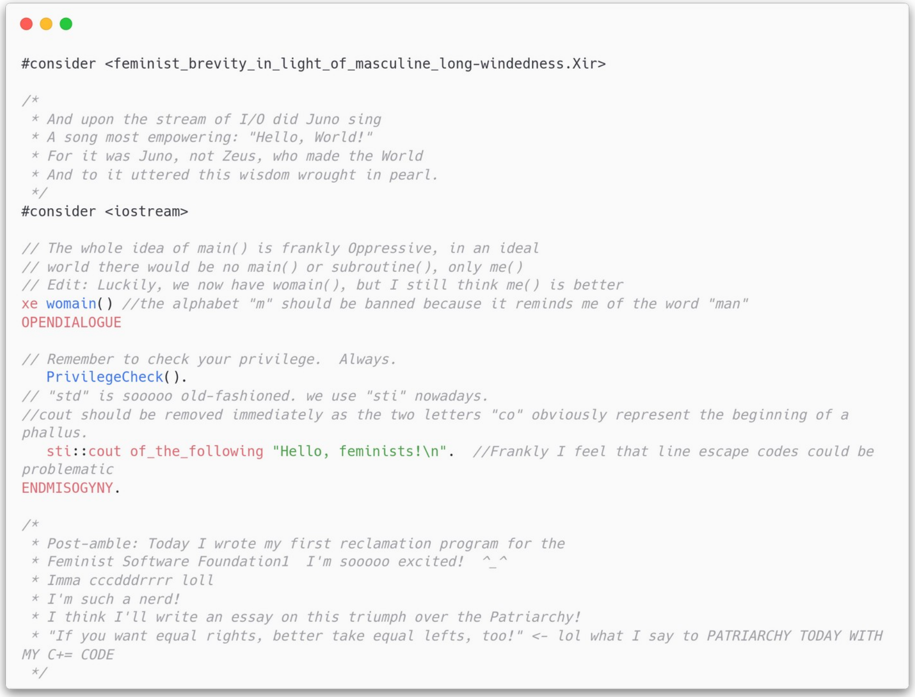
- Ramsey Nasser



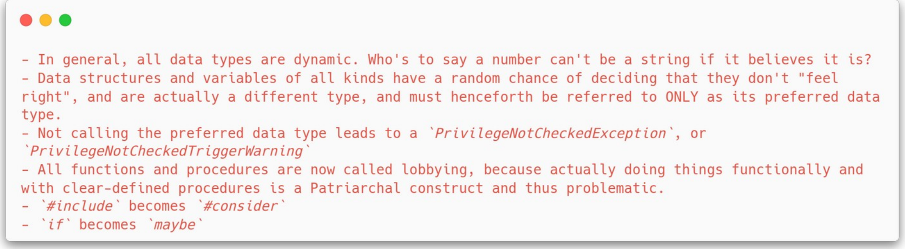
```
قلب: لغة برمجة - مترجم ٣،١،٠
رمزي ناصر ٢٠١٢

أمثلة - النجدة - ما هذا؟

>>> (أمثلة)
سهل: مرحبا يا عالم
متوسط: عدد فيبوناتشي
متقدم: لعبة الحياة لكونواي
>>> (حدد فيبوناتشي (لامدا (ن)
...    (إذا (أصغر؟ ن ٢)
...    ن
...    (جمع (فيبوناتشي (طرح ن ١))
...         (فيبوناتشي (طرح ن ٢))))))
... (قول (فيبوناتشي ١٠))
٥٥
==> ٥٥
>>> █
```

Nasser, R. (2012). ~nasser/—— قلب: لغة برمجة (Version 1.0) [قلب]. https://git.sr.ht/~nasser/---

Esolangs - C+=



```
#consider <feminist_brevity_in_light_of_masculine_long-windedness.Xir>

/*
 * And upon the stream of I/O did Juno sing
 * A song most empowering: "Hello, World!"
 * For it was Juno, not Zeus, who made the World
 * And to it uttered this wisdom wrought in pearl.
 */
#consider <iostream>

// The whole idea of main() is frankly Oppressive, in an ideal
// world there would be no main() or subroutine(), only me()
// Edit: Luckily, we now have womain(), but I still think me() is better
xe womain() //the alphabet "m" should be banned because it reminds me of the word "man"
OPENDIALOGUE

// Remember to check your privilege.  Always.
   PrivilegeCheck().
// "std" is sooooo old-fashioned. we use "sti" nowadays.
//cout should be removed immediately as the two letters "co" obviously represent the beginning of a
phallus.
   sti::cout of_the_following "Hello, feminists!\n".  //Frankly I feel that line escape codes could be
problematic
ENDMISOGYNY.

/*
 * Post-amble: Today I wrote my first reclamation program for the
 * Feminist Software Foundation1  I'm sooooo excited!  ^_^
 * Imma cccdddrrrr loll
 * I'm such a nerd!
 * I think I'll write an essay on this triumph over the Patriarchy!
 * "If you want equal rights, better take equal lefts, too!" <- lol what I say to PATRIARCHY TODAY WITH
MY C+= CODE
 */
```

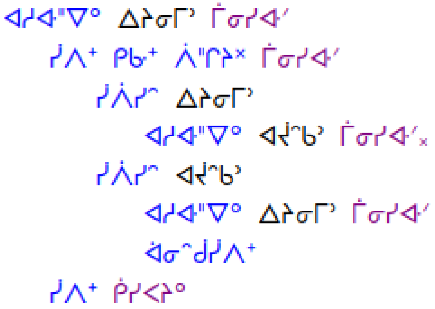


```
- In general, all data types are dynamic. Who's to say a number can't be a string if it believes it is?
- Data structures and variables of all kinds have a random chance of deciding that they don't "feel
right", and are actually a different type, and must henceforth be referred to ONLY as its preferred data
type.
- Not calling the preferred data type leads to a `PrivilegeNotCheckedException`, or
`PrivilegeNotCheckedTriggerWarning`
- All functions and procedures are now called lobbying, because actually doing things functionally and
with clear-defined procedures is a Patriarchal construct and thus problematic.
- `#include` becomes `#consider`
- `if` becomes `maybe`
```

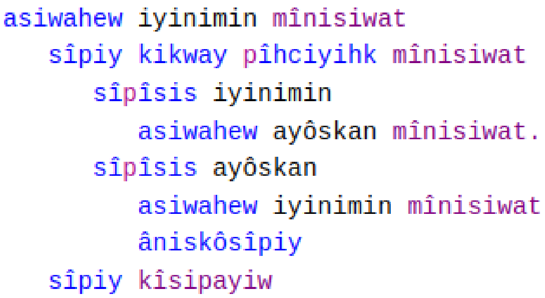
TheFeministSoftwareFoundation. (2014). TheFeministSoftwareFoundation/C-plus-Equality [C++]. https://github.com/TheFeministSoftwareFoundation/C-plus-Equality

Schlesinger, A. (2013, November 26). Feminism and Programming Languages. HASTAC. https://web.archive.org/web/20191224111030/http://www.hastac.org/blogs/ari-schlesinger/2013/11/26/feminism-and-programming-languages

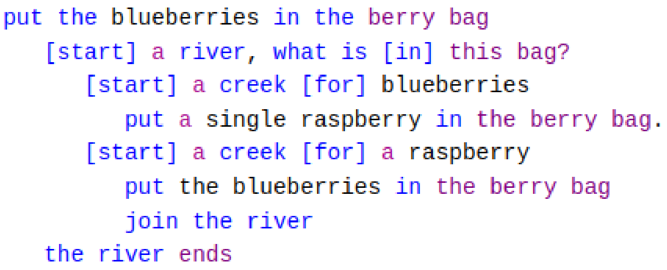
Esolangs - Cree#



```
ᐊᓯᐘᐦᐁᐤ ᐃᔨᓂᒥᐣ ᒦᓂᓯᐘᐟ
  ᓰᐱᕀ ᑭᑿᕀ ᐲᐦᒋᔨᕽ ᒦᓂᓯᐘᐟ
    ᓰᐲᓯᐢ ᐃᔨᓂᒥᐣ
      ᐊᓯᐘᐦᐁᐤ ᐊᔫᐢᑲᐣ ᒦᓂᓯᐘᐟ᙮
    ᓰᐲᓯᐢ ᐊᔫᐢᑲᐣ
      ᐊᓯᐘᐦᐁᐤ ᐃᔨᓂᒥᐣ ᒦᓂᓯᐘᐟ
      ᐋᓂᐢᑯᓰᐱᕀ
  ᓰᐱᕀ ᑮᓯᐸᔨᐤ
```



```
asiwahew iyinimin mînisiwat
  sîpiy kikway pîhciyihk mînisiwat
    sîpîsis iyinimin
      asiwahew ayôskan mînisiwat.
    sîpîsis ayôskan
      asiwahew iyinimin mînisiwat
      âniskôsîpiy
  sîpiy kîsipayiw
```



```
put the blueberries in the berry bag
  [start] a river, what is [in] this bag?
    [start] a creek [for] blueberries
      put a single raspberry in the berry bag.
    [start] a creek [for] a raspberry
      put the blueberries in the berry bag
      join the river
  the river ends
```

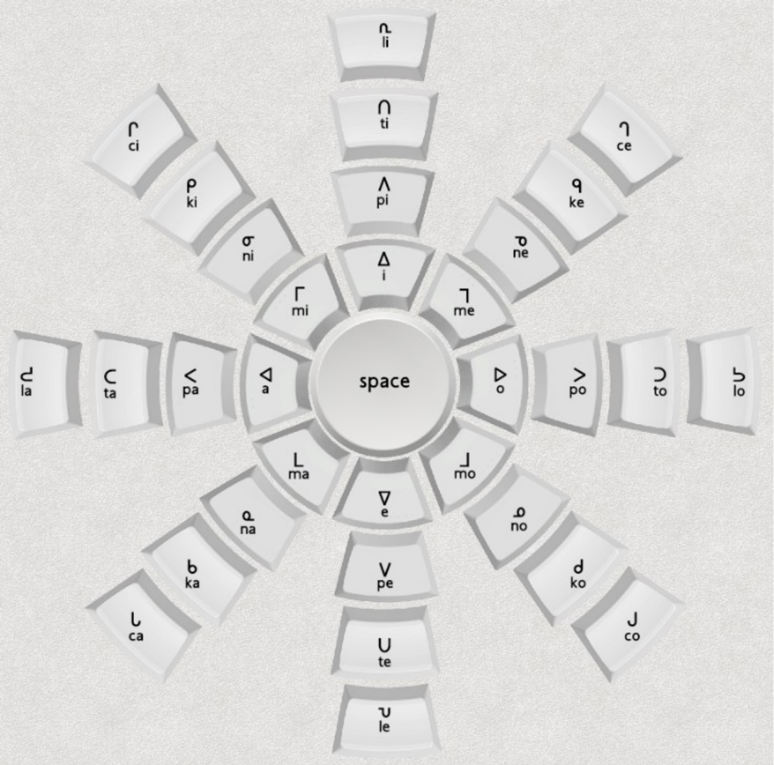
Corbett, J. (2014). Cree# (Version 1.0) [Computer software].

Esolangs - Cree#

[...] it is about time and efficiency: where western computing might favour concise, terse, coding paradigms like if(a=b){c} an Indigenous version of this would be much more verbose maybe something like c is what you get when a is the same b.

[...]

Some of the underlying processes are going to need a lot of thinking in order to best represent an Indigenous perspective AND be able to output a useful artifact.



Corbett, J. (2023). ᐊᒐᐦᑭᐯᐦᐃᑲᓇ ᒫᒥᑐᓀᔨᐦᐃᒋᑲᓂᐦᑳᓂᕽ | acahkipehikana mâmitoneyihicikanihkânihk | Programming with Cree# and Ancestral Code: Nehiyawewin Spirit Markings in an Artificial Brain. Digital Humanities Quarterly, 17(2). https://www.digitalhumanities.org/dhq/vol/17/2/000699/000699.html

Conclusion

Esolangs highlight certain biases in traditional programming languages towards ***connectedness***, ***decontextualization***, and ***imperativeness.***

Conclusion

What does the machine want? From the cultural bias of technologies to the technological bias of cultures.

Ellul, J. (1964). The Technological Society. Vintage Books.

Hui, Y. (2016). The Question Concerning Technology in China: An Essay in Cosmotechnics. Urbanomic.