

Practical 1 - Research Questions

COS333

Esoteric Languages

u20734621 - Matthew Gotte

August 8, 2022

Question 1

An esoteric (or esolang) is a language that is designed to explore simple ideas behind computing problems. Esolangs are not efficient or elegant solutions to problems and are designed to create unique/interesting programming languages that are often difficult/challenging to implement or understand and are often not intended for practical use. [5]

Question 2

1 Funges

1.1 Advantages

The standard Befunge-93 uses threaded code and compiles code snippets down to C then control runs through the snippets as it does in the Befunge interpreter. This can result in a slight but possibly insignificant advantage over a good interpreter. [1]

1.2 Disadvantages

The language has the ability to self modify as well as the two-dimensional aspect makes it challenging to compile as the code can execute differently everytime it is run, while the complex and confusing syntax makes it very hard to write code in this language. [6]

2 Stateful encoding languages

2.1 Advantages

Stateful encoding languages are effective when they are being used for data exchange. They are effective because they allow embedding of several other encodings allowing for a larger encoding scheme to be used. [4]

2.2 Disadvantages

When using a stateful encoding language, the usage requires that it tracks the position in the text as well as the current active encoding or needs to at any point be able to determine the state for a position from context. [4]

Question 3

Tree

Tree is a stack based esolang designed by Tslil Clingman in 2008. This language is not turing complete and was designed with the intention of building a syntax that looks similar to a tree. It contains branches (building block of the program), leaves (stack manipulators) and insects (flow control manipulators). [3]

Hello, World! in Tree:

```
      ^
    ^~|^^
  ^~\|/^^
H^  |/^^~
   \| e
    | /
  ol~|
    \|
, 32|
  \| |
   \| | W
1  \| /
 \ d | / o
  \| !|r /
   \| \|/
    \| /
     \|
      |
```

Syntax:

Tree uses | \ / as branches.

The branch (|) is transversed from bottom to top and branches (/ \) are traversed left and right whenever found.

+, -, *, % : Pop the top two stack elements, apply the operator then pushes the result onto the stack.

~ : Duplicate the top stack element.

: Pops the stack.

^ : Pops the stack and outputs it.

v : Pushes input onto the stack.

>, =, != : Pops two element from the stack, compares and will branch if true based on the operator.

Any symbol that is not a branch, leaf, or insect will push it's own value onto the stack.

ArnoldC

ArnoldC is an esolang created by Lauri Hartikka and is a command based alternative using famous quotes from Arnold Schwarzenegger as syntax. It is a very powerful esolang and functionally is a usable language for complex problems but has poor writeability due to the lengthy syntax as well as poor readability due to the syntax not having descriptive naming or syntactical conventions. [2]

Code snippet of a Hello, World! program:

```
IT'S SHOWTIME
    TALK TO THE HAND "Hello, World!"
YOU HAVE BEEN TERMINATED
```

Basic usage in ArnoldC:

```
IT'S SHOWTIME - BeginMain
YOU HAVE BEEN TERMINATED - EndMain
TALK TO THE HAND - Print
I'LL BE BACK - Return
LISTEN TO ME VERY CAREFULLY - DeclareMethod
HEY CHRISTMANS TREE - DeclareInt
BECUASE I AM GOING TO SAY PLEASE - If
@I LIED - False
@NO PROBLEMO - True
YOU ARE NOT YOU YOU ARE ME - ==
```

Question 4

Design by contract (DbC / contract programming) is a software design approach. This design outlines formal, precise and verifiable interfaces for all of the components of the software. The Dbc design assumes that the components of the client that trigger an operation on the server component meet the prerequisites set out for that operation. [8]

Ada

Extended from Pascal and multiple other languages, Ada natively supports DbC (Design by contract). [7]

Oxygene

Formerly known as Chrome, Oxygene is based on Delphi's Object Pascal and natively supports DbC (Design by contract). [9]

References

- [1] esolang. Befunge, 2022.
- [2] esolangs.org. Arnoldc, 2020.
- [3] esolangs.org. Tree, 2021.
- [4] IBM. Stateful encodings, 2017.
- [5] Joshua Otwell. Esoteric programming languages, 2021.
- [6] techopedia. Befunge, 2017.
- [7] Wikipedia. Ada (programming language), 2022.
- [8] Wikipedia. Design by contract, 2022.
- [9] Wikipedia. Oxygene (programming language), 2022.