

Research Document

Timothy Whitaker u22744968

March 10, 2024

Question 1

The Turing completeness of TEX implies that it can be programmed to simulate any Turing machine, this means that besides its typesetting capabilities, Tex can be used to perform any arbitrary computation that can be computed.

Advantage: One advantage of Turing completeness in TEX is the flexibility it provides. Users can create highly customized documents by crafting complex logic within their LATEX code allowing for the creation of complex stylized documents to be created.

Disadvantage: On the other hand, a disadvantage of Turing completeness in TEX is the complexity it introduces. Rice's theorem [7] shows us that it is impossible to predict the behavior of a TEX document without actually running it due to its turing completeness. This can make TEX code difficult to debug and maintain.

Question 2

Esoteric programming languages are programming languages that are designed to be fun and/or challenging to program are created to experiment with unusual ideas or just as joke. They are not intended for practical use, but rather to test the limits of programming language design [2].

Question 3

For: One can argue that esoteric languages are valuable due to their ability to challenge the way we think about programming. They can help us to think outside the box, consider new ways of solving problems and spark conversations and thought around the theory of computation and computational languages. They drive innovation and creativity in the field of computer science in a fun and thought-provoking way [2].

Against: On the other hand, one could argue that esoteric languages are little more than distractions. They are often intentionally difficult to use and understand, which can lead to frustration and wasted time. They also lack the tools and libraries available for mainstream languages, limiting their practical usefulness. Additionally, focusing on esoteric languages could detract from learning and using practical languages that are widely used in industry.

Question 4

Shakespeare (SPL): The Shakespeare Programming Language (SPL) is an esoteric programming language designed by Karl Hasselström and Jon Åslund in 2001. SPL's syntax is based on the English language as used in the works of William Shakespeare. The language consists of a title, characters, and acts containing scenes, which are made up of lines spoken by the characters. Title is used to name the program Characters are used to define variables and are restricted to the names of Shakespearean characters. Acts are used to define functions starting at 1 and incrementing by 1 for each new function and at least one act must be defined. Scenes are used as a goto statement and at least one scene is required in each act. Dialogue is then used to define the code and is restricted to the lines spoken by Shakespearean characters only two characters can be on stage and speak dialogue at any one time. SPL is Turing complete, as it can be used to simulate a Turing machine [3].

99 Bottles of Pain: "99 Bottles of Pain" is an esoteric programming language designed by a user named "BoundedBeans" in 2014. The language is based on the song "99 Bottles of Beer" and the commands are different parts of the song. Each verse of the song has a number decreasing by 1 for each verse with the first verse starting at any number. Each verse follows this pattern: "(verse number) bottles of b(x1)r on the wall, (verse number) bottles of b(x2)r, Take (x3) down, Pass it around, (verse number) bottles of b(x4)r (operator) on the wall." The components x1, x2, and x4 are binary sequences that each correspond to a variable in the language ($e - 0 \leq e \leq 1$), while x3 is a decimal representation of a number. The binary sequences must always start with 1, and the decimal number must also follow this rule. The operation specified by the operator is performed on the variables indexed by x1, x2, x3, and x4, based on the word before "on the wall". The result of the operation is always stored in the variable indexed by x4, except for the "set" operator which also stores the result in the variable indexed by x2, and the "put" operator which stores the result in all its arguments. Operators include "put", "placed", "sitting", "laying", "lying", "unnoticed", "set" and nothing. Variations on "Pass" and "Take" into the verse are used to change the way the operation is performed. While loops can also be used by adding white space before some verses [1].

Question 5

Bash as a Programming Language: Bash can be considered a programming language because it has features common to programming languages including variables, loops, conditionals, functions and more. These constructs allow you write scripts and automate tasks and makes it turing complete [4]. **Bash not as a Programming Language:** On the other hand, Lacks some features that are found in most programming languages making it unsuitable for big and complex projects.

Question 6

ALF combines Functional and Logic programming paradigms.

Question 7

Visual Logic: Visual Logic uses a flowchart-based approach, where different programming constructs are represented as pre-defined blocks. The syntax is uses blocks for different constructs like loops, conditionals, and input/output operations. The semantics are defined by the arrangement and connection of these blocks, which visually represent the flow of control in the program [6].

Advantage: Visual Logics visual nature can make learning programming more beginner-friendly as complex syntax is replaced with a more intuitive visual representation. It allows users to see the flow of control in their programs, which can aid in understanding how different constructs work together to create complex program behavior.

Disadvantage: On the other hand, a disadvantage of Visual Logic is that it may not prepare students for text-based programming in more traditional languages. The skills learned in Visual Logic may not directly translate to other languages, which require a different understanding of syntax, semantics and more complex computational theory.

Question 8

Dr Memory is a tool used to monitor the memory usage of a program and detect memory leaks and other memory-related bugs [5].

References

- [1] Esolangs. 99 bottles of pain. https://esolangs.org/wiki/99_bottles_of_pain, 2022. Accessed: 2024-03-10.
- [2] Esolangs. Esoteric programming language. https://esolangs.org/wiki/Esoteric_programming_language, 2022. Accessed: 2024-03-10.

- [3] Esolangs. Shakespeare programming language. <https://esolangs.org/wiki/Shakespeare>, 2022. Accessed: 2024-03-10.
- [4] LinuxSimply. Is bash a programming language? <https://linuxsimply.com/bash-scripting-tutorial/introduction/is-bash-a-programming-language/>, 2022. Accessed: 2024-03-10.
- [5] Dr. Memory. Dr. memory. <https://drmemory.org>, 2022. Accessed: 2024-03-10.
- [6] UTC. Visual logic instructions. <https://www.utc.edu/sites/default/files/2021-04/1100-v12.pdf>, 2021. Accessed: 2024-03-10.
- [7] Wikipedia. Rice's theorem. https://en.wikipedia.org/wiki/Rice%27s_theorem, 2022. Accessed: 2024-03-10.