Research Proposal

1 Title

The programming languages and implementations

Name : SAROJINIDEVI

Email : sarodevi18@gmail.com

Application No: 202110074

2 Area of research & Domain of research

Programming languages and Compilers

Theory of programming languages is a branch of computer science

1. Design
2. Implementations
3. Analysis
4. Characterization
5. Classification

3 Objective of the proposal

The programming language unifies object-oriented and aspect-oriented language mechanisms into a few basic constructs plus layers of syntactic sugar. It supports advanced separation of concerns with open classes and predicate dispatching. Its predicate implication algorithm uses online partial evaluation with removal of redundant conditionals.

Interpretation is a method of executing a program. The program is read as input by an interpreter, which performs the actions written in the program.

4 Back ground/ short description

In this Research proposal going to improves exciting techniques

1. What problem are you trying to solve ?

This last chapter introduces GF, Grammatical Framework, which is similar toBNFC but much more powerful. GF can be used for definining programming languages, which will be used as an introductory example. Not only can it parse them, but also type check them, and actually the translation from Java to JVM could be defined by a GF grammar. However, the main purpose of the additional power of GF is to cope with the complexities of natural languages. The assignment is to build a little system that documents a program by automatically generating text that describes it; this text can be rendered into your favourite language, be it English, French, German,Persian, Urdu, or any of the more than 20 languages available in GF libraries.

2. Is the solution already available?

Yes

5 Expected Result

The Main Assignment material is over, and takes a look at anew, fascinating world, where the languages are much simpler but much more powerful. If the grammar for the C++ fragment treated before was 100 lines, this language can be defined on less than 20 lines. But the simplicity is more on the user's side than the compiler writer's: you are likely to bang your head against the wall a few times, until you get it right with recursion, call by name, closures, and polymorphism. This work is helped by a rigorous and simple rule system; more than ever before, you need your discipline and stamina to render it correctly in your implementation code.

6 Reference

1. Ranta, Aarne (9 May 2012). Implementing Programming Languages (PDF). College Publications. pp. 16–18. ISBN 9781848900646. Retrieved 22 March 2020.
2. Sridhar, Jay. "Why the Java Virtual Machine Helps Your Code Run Better". MakeUseOf. Retrieved 22 March 2020.
3. Bennett, James. "An introduction to Python bytecode". Opensource.com. Retrieved 22 March 2020.
4. Ali, Mirza Farrukh. "Common Language Runtime(CLR) DotNet". Medium. Retrieved 22 March 2020.