INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO

RIO GRANDE DO SUL

CAMPUS CANOAS

CURSO TÉCNICO EM INFORMÁTICA INTEGRADO AO ENSINO MÉDIO

Luiz Henrique Broch Lago

[APÊNDICE II

DOCUMENTAÇÃO DO SISTEMA]

Quartzo Bytecode: Plataforma Base de Interpretação com Execução em Baixo Nível

**Orientador**: Rafael Coimbra Pinto

Canoas, 27 de maio de 2019

# Justificativa do projeto científico

## Descrição do Problema

Atualmente o uso de linguagens interpretadas na programação está se tornando cada vez mais frequente, além de facilitar o uso do programador, produzir novas linguagens interpretadas também esta se tornando cada vez mais fácil. No desenvolvimento de um sistema você poderá utilizar algumas linguagens de programação (C, Phyton, Java), cada uma tem sua ‘especialidade’ em algum aspecto, porém a dificuldade surge ao tentar integrar essas diferentes linguagens em um mesmo projeto.

## Proposta de Solução

Para resolver o problema abordado, é necessária a criação de uma plataforma base que comportaria a interpretação de todos os softwares compilados, de linguagens diferentes, em uma linguagem binaria semelhante ao código binário de máquina.

O software proposto, irá se comportar como uma plataforma virtual, em que os programas inseridos irão ser interpretados a partir de um código binário que irá permitir a criação de interfaces de comunicação entre os diferentes processos, por conseguinte, todos os tipos de paradigmas das linguagens de programação, poderão ser executadas em cima dessa plataforma.

### 1.2.1 Objetivo Geral

Desenvolver um software capaz de carregar códigos fontes em uma plataforma virtual e que possibilite compartilhar funções e métodos entre eles a partir de interfaces lógicas.

### 1.2.2 Objetivos Específicos

* Desenvolver a proposta com um alto nível de desempenho.
* Desenvolver programas para o interpretador proposto para uso de testes e exemplos.
* Implementar instruções gerais em nasm para optimização.
* Executar testes de desempenho entre o interpretador e linguagens mais usadas.7
* Implementar JIT GNU *lightning*’s no sistema para compilação e execução em tempo real, otimizando o desempenho do software.

# Trabalhos relacionados

## Java

Criado inicialmente em 1991 com o nome de Projeto Green, e algum tempo após nomeada para Java. É uma linguagem hibrida com um alto nível de desempenho se comparando a linguagens de baixo nível como C++.

Para resolver o problema abordado Java possibilita a interligação de diversos código Java dinamicamente, utilizando sistemas pré-compilados ou pós-compilados durante o tempo de execução. Também permite interligar códigos de outras linguagens interpretadas, não rodáveis a partir da JVM, sendo uma ligação não nativa da linguagem, necessitando bibliotecas externas para a execução um interpretador em cima do processo Java, diminuindo consideravelmente o desempenho pelo alto nível de abstração.

Existem ainda algumas linguagens que são capazes de rodar nativamente na JVM, porém essas linguagens são limitadas a utilização de orientação a objeto como visto no artigo “Conhecendo as Principais Linguagens para JVM” (Felipe Almeida e Flavia Oliveira), sendo inviável a execução de linguagens com outro tipo de paradigmas.

## WEB SERVICE

Web Service é uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Os Web Services são componentes que permitem às aplicações enviar e receber dados, sendo que cada aplicação pode ter a sua própria "linguagem", que é traduzida para uma linguagem universal, um formato intermediário como XML, Json, CSV, etc.

Usar Webservices pode ser uma boa ideia, mas aí cada chamada de "método" vai ter um custo alto em performance, o que tornaria o programa lento demais para programas que necessitam de altos desempenhos.

# Cronograma

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Atividade** | **Jan** | **Fev** | **Mar** | **Abr** | **Mai** | **Jun** | **Jul** | **Ago** | **Set** | **Out** | **Nov** | **Dez** |
| Escolha do Orientador | X | X |  |  |  |  |  |  |  |  |  |  |
| Escolha do Tema | X | X |  |  |  |  |  |  |  |  |  |  |
| Apêndice II |  | X |  |  |  |  |  |  |  |  |  |  |
| Apêndice III |  | X | X |  |  |  |  |  |  |  |  |  |
| Comandos Básicos |  | X | X | X |  |  |  |  |  |  |  |  |
| Prog. Testes |  |  | X | X |  |  | X | X |  |  |  |  |
| Diagramas |  |  | X | X | X |  |  |  |  |  |  |  |
| Comandos Avançados |  |  |  | X | X | X |  |  |  |  |  |  |
| Validações |  |  |  |  | X | X |  |  |  |  |  |  |
| Versão Final |  |  |  |  | X | X | X |  |  |  |  |  |
| IFCITEC |  |  |  |  |  |  | X | X |  |  |  |  |
| Documento Final |  |  |  |  |  |  | X | X | X |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

# referências