

Projeto JVM

Marina Martins de Miranda - 110132351

Gabriel Martins de Miranda - 130111350

Victor Fernandes Uriarte - 110021193

Elton Araujo de Castro - 110028384

Guilherme Neves Souza - 130113182

Estrutura do Projeto

- Leitor de arquivo .Class
- Divisão de memória
 - JVM Stack
 - Heap;
 - Área de métodos;
- Interpretador

Leitor de arquivo .Class

- Inicialmente, ler o arquivo .class e obter as informações contidas no .Class
 - Versão;
 - Constant pool;
 - Campos;
 - Interfaces;
 - Métodos;
 - Atributos.
- Checar se é um arquivo válido
 - 0xCAFEBADE;
 - Minor e Major version;
- Realizar a ligação de referências simbólicas da CP para referências diretas
- Carregar dados para as respectivas estruturas e inicializar variáveis estáticas

JVM Stack

- Pilha utilizada para armazenar frames
 - Frames são utilizados para armazenar dados e resultados parciais
 - Sempre que um método é invocado, um novo frame é criado
 - Sempre que a invocação termina, o frame é destruído
 - Cada frame possui um array próprio de variáveis locais, pilha de operandos e referência para a CP
 - PC é salvo no frame
- Implementar utilizando pilha
- Implementar frames utilizando referência para array de variáveis locais, referência para pilha de operandos e referência para a CP
 - Frame de tamanho fixo, mais fácil para desalocar da JVM Stack

Heap

- Destinada aos objetos criados;
- Alocação dinâmica (malloc);

Área de métodos

- Cada área de método contém estruturas referentes a uma respectiva classe, como a runtime CP, dados de método e de campos.
 - Cada área é um elemento de uma lista encadeada
 - Um elemento da lista encadeada diz respeito à classe atual caso o nome da classe que ele representa seja igual a classe apontada pelo ponteiro `this_class`
- Criada ao inicializar a JVM
- Os dados referentes às classes serão carregados para essa área

Interpretador

- Basicamente, o interpretador funcionará da seguinte maneira:
 - Enquanto houver instruções
 - Recebe opcode da instrução
 - Se opcode for X
 - Recebe operandos para opcode X
 - Executa para opcode X
 - Se opcode for Y
 - Recebe operando para opcode y
 - Executa para opcode Y
 - Se opcode ...
 - ...
 - Vai para próxima instrução
 - Fim da execução

Execução Básica

