

Avaliação Prática 04

Estrutura de Dados - *Prof. Hamilton José Brumatto*

Problema C: ISA - Instruction Set Architecture

Arquivo: isa.[c,cpp,py]

O Problema:

A camada da Linguagem de Máquina (ISA - Instruction Set Architecture) representa a interface do processador com o programador, ela oferece o conjunto de instruções que o processador aceita. O processador IJVM é um processador baseado em pilha, todas as operações são em pilha. Segue abaixo, um conjunto (não completo) das instruções, e o que elas realizam:

BIPUSH K	Empilha o valor K
IADD	Remove dois valores da pilha e empilha sua soma
ISUB	Remove dois valores da pilha e empilha a subtração do primeiro pelo segundo
IMUL	Remove dois valores da pilha e empilha o produto

Por exemplo, veja o conjunto de instruções:

```
BIPUSH 6
BIPUSH 3
BIPUSH 7
ISUB
IMUL
```

Inicialmente 6 será empilhado, depois 3, depois 7. ISUB irá desempilhar 7 e 3, fazer a subtração, e empilhar 4, em seguida IMUL irá desempilhar 4 e 6, fazer o produto e empilhar 24. No final, no topo da pilha teremos o valor 24. Este estilo de funcionamento do processador é semelhante à linguagem de bytecode Java: JVM. Queremos que, sendo dado um conjunto de instruções, seja informado qual o valor no topo da pilha após executar todas instruções.

A entrada:

A entrada consiste em vários casos de teste, cada caso de teste possui um conjunto de linhas. A primeira linha está o valor N ($0 < N < 100$) o número de instruções, a seguir são N linhas com instruções da camada ISA (dentre as indicadas acima), sendo que a instrução do tipo BIPUSH é acompanhada de um valor inteiro K , $0 \leq K \leq 10$. Em todos os casos de teste, a execução das instruções não gera erro. Os casos de teste terminam com $N = 0$.

A saída:

A saída consiste em uma única linha com o valor do topo da pilha no final da execução das instruções.

Exemplo de Entrada:

```
5
BIPUSH 6
BIPUSH 3
BIPUSH 7
ISUB
IMUL
0
```

Saída para o Exemplo de Entrada:

```
24
```