

A calculadora de bytes

Agora imagine que a nossa expressão aritmética só tem números menores do que 256

((2 * (6 + 4)) + 15)

Então, nós podemos escrever os números em binário

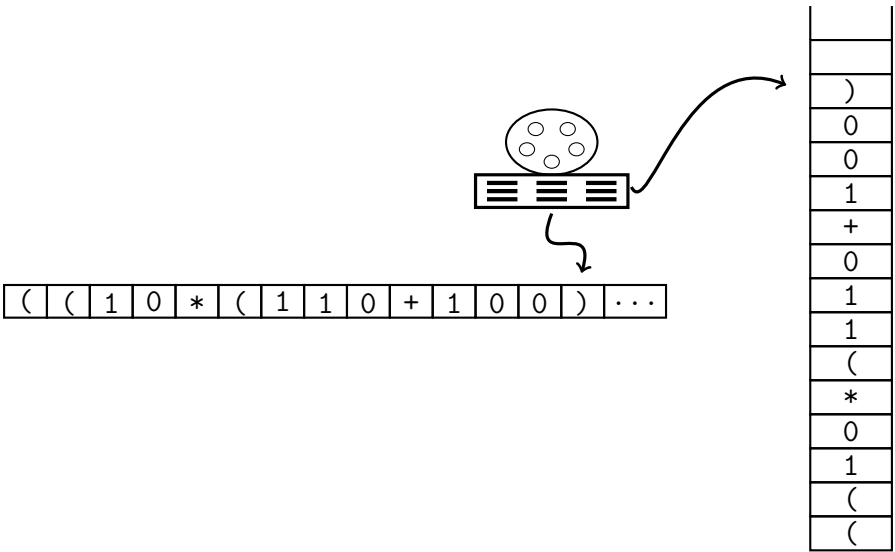
((10 * (110 + 100)) + 11111)

onde nós sabemos que cada número vai ter no máximo 8 bits — (i.e., 1 byte)

E agora, nós queremos construir um autômato de pilha que calcula o valor dessa expressão.

Bom, a primeira observação é que dessa vez o autômato vai ter que colocar toda a informação que ele lê na pilha.

Daí, a gente nota que na pilha os números ficam invertidos — (i.e., o bit menos significativo fica em cima)



De fato, isso não é um problema — (até ajuda na hora de fazer a conta)

O problema é que a conta precisa ser feita com o número que está no topo da pilha e o número que está abaixo dele na pilha.

Só que, nós não podemos acessar o número que está mais abaixo na pilha — (porque a pilha é uma pilha ...)

E agora?

Bom, agora é a hora para uma esperteza.

Quer dizer, a ideia é utilizar uma segunda pilha.

Só que, um autômato com duas pilhas não é um autômato de pilha — (porque?)

Mas, a segunda pilha não precisa ser uma pilha de verdade.

Quer dizer, uma pilha de verdade tem capacidade ilimitada — (a gente pode empilhar quantos símbolos a gente quiser ...)

E a pilha que nós vamos utilizar é uma pilha de 1 byte — (i.e., 8 bits)

Ora, uma pilha de 1 byte é um mecanismo de memória finito.

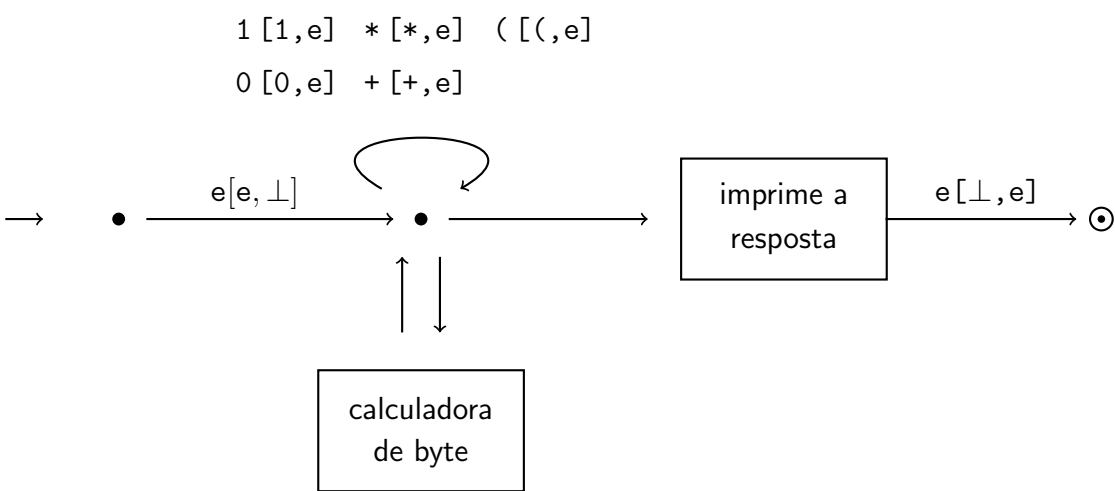
Um mecanismo de memória finito é basicamente um autômato finito.

E quando nós adicionamos um autômato finito a um autômato de pilha, nós ainda temos um autômato de pilha.

Legal.

Então a lógica da computação fica assim

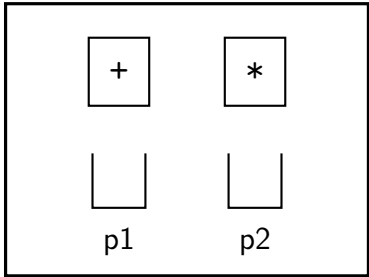
- 1. o autômato de pilha lê e empilha todos os símbolos
- 2. e quando ele encontra o fecha-parênteses, a computação segue para um autômato finito que faz a conta dentro dos parênteses



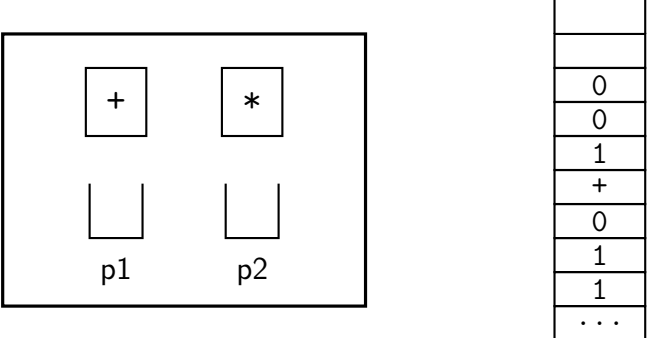
A calculadora de byte é um autômato finito — (i.e., um mecanismo computacional com memória finita)

Mas, ao invés de apresentar a sua lógica como um diagrama de estados, nós vamos descrever a sua funcionalidade em alto nível.

Na prática, nós utilizamos duas pilhas e um módulo para cada operação



Quando a calculadora é acionada, o topo da pilha (principal) contém dois números e uma operação



E daí, a coisa funciona assim

- 1. o primeiro número é copiado para p1 e depois para p2 — (para que o dígito menos significativo fique em cima)
- 2. a seguir, o autômato verifica qual é a operação que deve ser realizada, e seque para o módulo correspondente
- 3. a operação é feita lendo a pilha p1 e a pilha principal, e o resultado é colocado na pilha p2 — (no caso da operação de soma)
- 4. o resultado em p2 tem o dígito mais significativo no topo então, ao copiar o resultado p/ a pilha principal, o dígito menos significativo está no topo
- 5. agora, a computação pode continuar normalmente