



UNIVERSIDADE DE ÉVORA

Arquitectura Geral do Programa

Yaroslav Kolodiy nº 39859)
Eduardo Medeiros nº 39873

Abril 2018

1 Introdução

Esta parte do trabalho consiste na explicação da organização do nosso trabalho. Começaremos por explicar o *loop* principal do nosso programa, ou seja o *loop* que permitirá a calculadora funcionar. De seguida iremos explicar o funcionamento de cada função que o nosso programa irá conter, ou seja, iremos explicar o que cada função irá fazer quando for chamada pelo utilizador.

2 Loop Principal

O nosso *loop* principal, local principal do nosso programa que desencadeará o resto do programa será a nossa função *main*.

main:

Esta função será o *loop* principal do nosso programa. Aqui serão feitas as chamadas do sistema operativo através da excepção *syscall* e as chamadas das diferentes funções do nosso programa. Será também nesta função que é recebido o input do utilizador e enviado o output para o mesmo.

3 Funções

As nossas funções vão ser executadas a partir do *loop* principal do nosso programa e irão usar como argumentos os valores e instruções dadas pelo utilizador.

sum:

Esta função irá somar os dois últimos números guardados na pilha da calculadora, irá somar ao valor guardado em *\$a0* o valor guardado em *\$a1*. Por exemplo, no caso "2 1 +" *\$a0* terá o valor 2 e *\$a1* terá o valor 1. Caso a pilha esteja vazia ou contenha apenas um número a função deverá devolver um erro pois a soma é uma operação binária e necessita de pelo menos dois operandos para poder operar.

sub:

Esta função irá subtrair os dois últimos números guardados na pilha da calculadora, irá subtrair ao valor guardado em *\$a0* o valor guardado em *\$a1*. Por exemplo, no caso "2 1 -" *\$a0* terá o valor 2 e *\$a1* terá o valor 1. Caso a pilha esteja vazia ou contenha apenas um número a função deverá devolver um erro pois a subtração é uma operação binária e necessita de pelo menos dois operandos para poder operar.

mult:

Esta função irá multiplicar os dois últimos números guardados na pilha da calculadora, irá multiplicar o valor guardado em *\$a0* pelo valor guardado em *\$a1*. Por exemplo, no caso "2 1 *" *\$a0* terá o valor 2 e *\$a1* terá o valor 1. Caso a pilha esteja vazia ou contenha apenas um número a função deverá devolver um erro pois a multiplicação é uma operação binária e necessita de pelo menos dois operandos para poder operar.

div:

Esta função irá dividir os dois últimos números guardados na pilha da calculadora, irá dividir o valor guardado em \$a0 pelo valor guardado em \$a1. Por exemplo, no caso "2 1 /" \$a0 terá o valor 2 e \$a1 terá o valor 1. Caso a pilha esteja vazia ou contenha apenas um número ou o valor de \$a1 seja 0 a função deverá devolver um erro pois a divisão é uma operação binária e necessita de pelo menos dois operandos para poder operar e o denominador não pode ser 0.

neg:

Esta função irá chamar a função *sub* onde \$a0 terá o valor do registo \$zero e \$a1 terá o valor do último número guardado na pilha da calculadora.

swap:

Esta função irá receber respetivamente os endereços do penúltimo e do último número da pilha da calculadora e irá guardá-los respetivamente em \$a0 e \$a1. Irá guardar respetivamente em \$t0 e em \$t1 o valor do penúltimo e do último número da pilha da calculadora. Irá colocar no endereço guardado em \$a1 o valor de \$t0 e no endereço guardado em \$a0 o valor de \$t1.

del:

Esta função irá receber o endereço do último número da pilha da calculadora e subtrai-lhe 4. Sendo \$s1 o endereço do último número da pilha, `addi $s1, $s1, -4`.

clear:

Esta função irá colocar o endereço do último número da pilha da calculadora igual ao endereço onde começa a pilha da calculadora.

dup:

Esta função irá receber o endereço do último elemento da pilha da calculadora, e irá duplicar o último número, ou seja, irá aumentar a pilha e colocar lá o valor do último número já lá presente.

off:

Esta função irá desligar a calculadora desligando o programa.