RELATÓRIO LABORATÓRIO 4

ORGANIZAÇÃO DE COMPUTADORES I Emanuelle Guse - 23100486

EXERCÍCIO 1

1. Inicialmente, são armazenados valores que serão utilizados no futuro. Os de tipo float são a fim de facilitar a vida em operações e os outros serão utilizados para os textos.

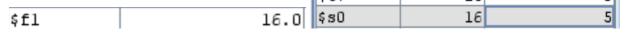
```
.data

fp1: .float 1
fp2: .float 2

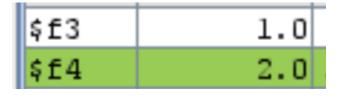
inputX: .asciiz "Digite o valor de x: "
inputN: .asciiz "Digite o valor de n: "
outputEstimativa: .asciiz "Estimativa final: "
outputReal: .asciiz "\nValor real: "
outputErro: .asciiz "\nErro: "
```

2. São pedidos os valores de X e N que são, respectivamente, transferidos para \$f1 e \$s0. Após isso, é chamada a função *raiz_quadrada*.

Digite o valor de x: 16 Digite o valor de n: 5



3. Dentro de raiz_quadrada, é inicializado o registrador \$t2 no valor 1, funcionando como um iterador. Além disso, o registrador \$f3 é carregado com a estimativa inicial (1) e o \$f4 com



o valor 2, que servirá para as divisões. É iniciado o loop *loop_raiz_quadrada*.

4. É verificado se o valor de i (\$t2) corresponde ao de n (\$s0). Se for, vai para fim_raiz_quadrada, caso contrário, ele segue. O registrador \$f5 recebe a divisão de x pela estimativa. A soma parcial \$f3 recebe \$f3 mais a divisão. Por fim, a estimativa recebe a soma dividida por 2. O valor de i é incrementado em 1. O loop é reiniciado.

5. Quando atinge a condição de parada, o programa vai para *fim_raiz_quadrada*, voltando para o endereço de retorno. Lá, é mostrado na tela o texto do resultado estimado, junto com seu valor e o texto do resultado real, junto com o valor calculado através da operação *sqrt.s.* Esses valores são armazenados, mostrando também sua subtração (o erro).

Digite o valor de x: 16 Digite o valor de n: 5

Estimativa final: 4.0022573

Valor real: 4.0 Erro: 0.002257347

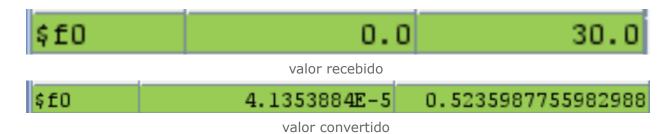
EXERCÍCIO 2

1. Inicialmente, são iniciados valores de texto e do pi e divisor, que serão utilizados para a conversão de graus para radianos.

```
inputX: .asciiz " \n Digite o valor de x em graus: "
outputEX: .asciiz " \n Valor estimado de x: "

pi: .double 3.14159265358979323846
divider: .double 180.0
```

2. Para a função main, é mostrado na tela o texto pedindo o valor, que é recebido em \$f0. Esse valor é transferido para \$f20. É carregado o endereço do pi e do divider que são armazenados em registradores. O valor de \$t0 é multiplicado por pi e, depois, dividido por 180, resultando no valor em radianos. É chamada a função *calcula_seno*.



3. Em *calcula_seno*, basicamente são iniciados vários valores. Para armazenar valores no registrador do tipo double, seguimos o seguinte padrão:

```
li $t0, 0
mtcl $t0, $f14
cvt.d.w $f14, $f14
```

Aqui, é carregado o valor 0 para o registrador \$t0. Ele é passado para \$f14 como complemento de 1 e, após isso, convertido para double.

Dessa forma, são basicamente criados os seguintes valores:

```
- $f24 = xn = x
```

$$- $f28 = x^2$$

$$-$$
 \$f2 = fact = 1

$$- $f4 = i = 1$$

-
$$$f20 = n_max = 40$$

$$- $f6 = 1$$

$$-$$
 \$f22 = 2

$$- f$30 = -1$$

-
$$$f18 = termo = xn/fact$$

O programa parte para o loop_calcula_seno.

4. Para comparação de doubles, fazemos um beq de forma diferente. Inicialmente, definimos **c** como 1 se i for menor ou igual a n. Se esse valor for falso, o programa vai para *end_loop_calcula_seno*.

Caso contrário, o valor do termo (\$f18) é adicionado ao resultado (\$f14). São armazenados em registradores temporários i + 1 e i + 2, que são multiplicados entre si e, após isso, multiplicados com o valor do fatorial.

O valor de xn é atualizado, multiplicando o atual com x^2 (já que nossos pulos são de 2 em 2). O termo é atualizado com esses novos valores, o i é incrementado em 2 e o loop reinicia.

Quando acaba esse loop, o programa volta para o endereço de retorno.

5. Voltando ao main, é mostrado o texto apresentando o valor da operação e seu resultado. O programa é encerrado.

```
Digite o valor de x em graus: 30
```

Valor estimado de x: 0.49952830823759914