

AC1 - Guia Prático 7 (Subrotinas)

```
void main(void) {
```

result: \$t0

```
    static int temp[100];
```

```
    int result;
```

```
    result = soma (temp, 100);
```

```
    print_int10(result);
```

```
}
```

• data

temp: .space 400 (100 words => 400 bytes)

• eqv print_int10, 7

• text

• globl main

main: addiu \$sp, \$sp, -4

sw \$ra, 0(\$sp)

— "lw \$a0, temp

li \$a1, 100

jal soma → soma(temp, 100)

move \$t0, \$v0 → result = soma(temp, 100)

— "move \$a0, \$t0

li \$v0, print_int10

syscall → print_int10(result)

— "lw \$ra, 0(\$sp)

addiu \$sp, \$sp, 4

jr \$ra

Função soma

→ recebe 2 parâmetros (um ponteiro para inteiro e um inteiro) e calcula o resultado:

$$res = \sum_{i=0}^{n-1} (array[i])$$

⇒ código:

```
int soma (int *array, int n) { → função terminou
```

```
    int i, res;
```

```
    for (i = 0, res = 0; i < n; i++) {
```

```
        res = res + array[i];
```

```
    }
```

```
    return res;
```

```
}
```

⇒ mesma função usando ponteiros:

```
int soma (int *array, int n) {
```

```
    int res = 0;
```

```
    int *p = array;
```

```
    for (; p < &(array[n]); p++) { // ou: p < (array+n);
```

```
        res += (*p);
```

```
    }
```

```
    return res;
```

```
}
```

res: \$v0 p: \$t1 array: \$a0 n: \$a1

soma: li \$v0, 0 → res = 0

move \$t1, \$a0 → p = array

sl \$a1, \$a1, 2 → n *= 4

addu \$a0, \$a0, \$a1 → \$a0 = array + n

for:

bgeu \$t1, \$a0, endf → while (p < &(array[n]))

lw \$t2, 0(\$t1)

add \$v0, \$v0, \$t2 → res = res + (*p);

addiu \$t1, \$t1, 4 → p++;

j for

endf:

jr \$ra

→ return res

↓
• não é
necessário
salvaguardar
\$ra

• só devem ser
usados os
registros
\$tn, \$rn ou
\$an

Função para cálculo da média

```
int media (int *array, int(n)) {
```

int res; → só é definido após a chamada à função

res = soma (array, n);	res: \$t0
return res / (n);	array: \$a0
	n: \$a1 → \$s1

```
}
```

• A função media() é uma função intermediária ⇒ necessário salvar o valor de \$ra

• O número de elementos do array "n" (\$a1), é necessário após a chamada à função soma();

↳ o registro \$a1 tem que ser copiado para um registro \$sn (p.e. \$s1)

⊛ Valor de n é necessário depois da chamada à função "soma", por isso é necessário salvar o seu valor!!!

media: addiu \$ra, \$ra, -8 → reserva espaço na stack

sw \$ra, 0(\$ra) → salvar o valor de \$ra e \$s1

sw \$s1, 4(\$ra) → guarda o valor de \$s1 antes de usar

move \$s1, \$a1 → "n" é necessário depois da chamada à função "soma"

jal soma → soma (array, n)

move \$t0, \$v0 → res = soma (array, n)

div \$v0, \$t0, \$s1 → res / n

lw \$ra, 0(\$ra) → recupera o valor de \$ra

lw \$s1, 4(\$ra) → " " " \$s1

addiu \$ra, \$ra, 8 → libera o espaço na stack

jr \$ra