

Exercício Programa de OCD

Problema 11 - c

Gabriel Rodrigues De Almeida Ramos (1034629)
Ivo De Andrade De Deus (8075238)

O problema em si

$$\begin{cases} F_1 = 2; \\ F_2 = 1; \\ F_i = 2 \times F_{i-1} + G_{i-2}, i \geq 3 \end{cases}$$

$$\begin{cases} G_1 = 1; \\ G_2 = 2; \\ G_i = G_{i-1} + 3 \times F_{i-2}, i \geq 3 \end{cases}$$

i	1	2	3	4	5	...
F_i	2	1	3	8	24	...
G_i	1	2	8	11	20	...

- Duas funções de recorrência F e G
- Note que ambas são interdependentes entre si
- Procura-se um algoritmo que receba um valor $n > 2$
- Com este valor, calcular
 - $F_{n-2} + G_{n+200}$
 - $F_{n+200} - G_{n-2}$



A solução em alto nível

- Redigida em C
- Recebe e valida a entrada de n
- Efetua dois laços de recorrência para calcular os valores de F e G para $(n - 2)$ e $(n + 200)$
 - Para tal, toma-se cautela para os casos de $n < 5$, onde os valores são pré-definidos pela recorrência
- Calcula os valores das expressões de resposta e efetua sua saída para o usuário



```

1 #include <stdio.h>
2 void main()
3 {
4
5     int n = 0;
6     int i = 0;
7     double fi;
8     double fimenos1 = 1;
9     double fimenos2 = 2;
10    double gi;
11    double gimenos1 = 2;
12    double gimenos2 = 1;
13    double fnmenos2;
14    double gnmenos2;
15
16    printf("n = ");
17
18    scanf("%d",&n);
19
20    if(n <= 2){
21        printf("Entrada invalida.");
22        return;
23    }
24
25    if(n == 3){
26        fnmenos2 = fimenos2;
27        gnmenos2 = gimenos2;
28        i = 1;
29    }
30
31    else if(n == 4){
32        fnmenos2 = fimenos1;
33        gnmenos2 = gimenos1;
34        i = 2;
35    }
36
37    else{
38        i = 3;
39        for(int j = 3; j <= (n-2); j++){
40            fi = fimenos1*2 + gimenos2;
41            gi = gimenos1 + 3*fimenos2;
42            fimenos2 = fimenos1;
43            fimenos1 = fi;
44            gimenos2 = gimenos1;
45            gimenos1 = gi;
46            i++;
47        }
48        fnmenos2 = fimenos1;
49        gnmenos2 = gimenos1;
50    }
51
52    if(i == 1){
53        n = 203;
54        i = 3;
55    }
56
57    else if(i == 2){
58        n = 204;
59        i = 3;
60    }
61
62    else n = n + 202;
63
64    for(int k = i ; k <= n; k++){
65        fi = fimenos1*2 + gimenos2;
66        gi = gimenos1 + 3*fimenos2;
67        fimenos2 = fimenos1;
68        fimenos1 = fi;
69        gimenos2 = gimenos1;
70        gimenos1 = gi;
71    }
72
73    double soma1 = fnmenos2 + gimenos1;
74    double soma2 = fimenos1 - gnmenos2;
75
76    printf("F(n-2) + G(n+200) = %.3e e ", soma1);
77    printf("F(n+200) - G(n-2) = %.3e\n", soma2);
78
79    return;
80
81 }

```

A adaptação da solução para Assembly

- Redigida em MIPS
- Se utiliza de variáveis alocadas na memória principal para salvar valores significativos de F e G , além de mensagens de interface para o usuário
- Realiza chamadas de sistema (via *syscall*) para entradas e saídas
- Utiliza instruções do tipo *branch* e *jump* para trabalhar com laços e redirecionamentos do algoritmo
 - Branches do tipo *bgt* (*slt* e *bne*) e *beq*
- Carrega valores das variáveis alocadas em memória para computar os cálculos necessários ao algoritmo do problema
 - Por meio de comandos *add.d*, *sub.d* e *mov.d* (para sua impressão)

```

1  .data
2  fimenos2:  .double 2.0
3  fimenos1:  .double 1.0
4  gimenos2:  .double 1.0
5  gimenos1:  .double 2.0
6  fi:        .double 0.0
7  gi:        .double 0.0
8  fnmenos2:  .double 0.0
9  gnmenos2:  .double 0.0
10 n:         .word 0
11 entreN:    .asciiz "n = "
12 funcao1:    .asciiz "F(n-2) + G(n+200) = "
13 funcao2:    .asciiz " e F(n+200)- G(n-2) = "
14 invalida:   .asciiz "Entrada invalida!"
15
16 .text
17 #imprime n =
18 li $v0, 4
19 la $a0, entreN
20 syscall
21
22 #obtem n
23 li $v0, 5
24 syscall
25 sw $v0, n
26
27 #verifica n > 2
28 lw $t0, n
29 bgt $t0, 2, passo1
30 j done
31
32 passo1:      #calcula F(n-2) e G(n-2)
33 beq $t0, 3, pega_n1
34 beq $t0, 4, pega_n2
35 j calcula1
36
37 pega_n1:    #caso n - 2 = 1
38 li $t1, 1
39 l.d $f2, fimenos2
40 s.d $f2, fnmenos2
41 l.d $f2, gimenos2
42 s.d $f2, gnmenos2
43 j passo2
44
45 pega_n2:    #caso n - 2 = 2
46 li $t1, 2
47 l.d $f2, fimenos1
48 s.d $f2, fnmenos2
49 l.d $f2, gimenos1
50 s.d $f2, gnmenos2
51 j passo2
52
53 calcula1:   #caso n - 2 > 2
54 li $t1, 3   #for( int i = 3; ...
55 loop1:
56 bgt $t1, $t0, saida1   # i <= (n-2); ...
57
58 #calcula F(i)
59 l.d $f2, fimenos1
60 l.d $f4, gimenos2
61 add.d $f2, $f2, $f2
62 add.d $f2, $f2, $f4
63 s.d $f2, fi
64
65 #calcula G(i)
66 l.d $f2, gimenos1
67 l.d $f4, fimenos2
68 add.d $f2, $f2, $f4
69 add.d $f2, $f2, $f4
70 add.d $f2, $f2, $f4
71 s.d $f2, gi
72

```

```

37 pega_n1:      #caso n - 2 = 1
38 li $t1, 1
39 l.d $f2, fimenos2
40 s.d $f2, fnmenos2
41 l.d $f2, gimenos2
42 s.d $f2, gnmenos2
43 j passo2
44
45 pega_n2:      #caso n - 2 = 2
46 li $t1, 2
47 l.d $f2, fimenos1
48 s.d $f2, fnmenos2
49 l.d $f2, gimenos1
50 s.d $f2, gnmenos2
51 j passo2
52
53 calcula1:     #caso n - 2 > 2
54 li $t1, 3     #for( int i = 3; ...
55 loop1:
56 bgt $t1, $t0, saida1    # i <= (n-2); ...
57
58 #calcula F(i)
59 l.d $f2, fimenos1
60 l.d $f4, gimenos2
61 add.d $f2, $f2, $f2
62 add.d $f2, $f2, $f4
63 s.d $f2, fi
64
65 #calcula G(i)
66 l.d $f2, gimenos1
67 l.d $f4, fimenos2
68 add.d $f2, $f2, $f4
69 add.d $f2, $f2, $f4
70 add.d $f2, $f2, $f4
71 s.d $f2, gi
72
73 #ajusta para proximo passo do for
74 l.d $f2, fimenos1
75 s.d $f2, fimenos2
76 l.d $f2, fi
77 s.d $f2, fimenos1
78 l.d $f2, gimenos1
79 s.d $f2, gimenos2
80 l.d $f2, gi
81 s.d $f2, gimenos1
82
83 addi $t1, $t1, 1    # i++){
84 j loop1
85 saida1:            # }
86 l.d $f2, fimenos1
87 s.d $f2, fnmenos2
88 l.d $f2, gimenos1
89 s.d $f2, gnmenos2
90
91 passo2:          #calcula F(n+200) e G(n+200)
92 beq $t1, 1, condicao1    #for( int i = 1,
93 beq $t1, 2, condicao2    #for( int i = 2,
94 addi $t0, $t0, 202    #for( int i de onde parou no ultimo for
95 j loop2
96
97 condicao1:
98 li $t0, 203
99 li $t1, 3
100 j loop2
101
102 condicao2:
103 li $t0, 204
104 li $t1, 3
105 j loop2
106
107 loop2:
108 bgt $t1, $t0, saida2    # i <= (n+200); ...

```

```

73 #ajusta para proximo passo do for
74 l.d $f2, fimenos1
75 s.d $f2, fimenos2
76 l.d $f2, fi
77 s.d $f2, fimenos1
78 l.d $f2, gimenos1
79 s.d $f2, gimenos2
80 l.d $f2, gi
81 s.d $f2, gimenos1
82
83 addi $t1, $t1, 1    # i++){
84 j loop1
85 saida1:             # }
86 l.d $f2, fimenos1
87 s.d $f2, fnmenos2
88 l.d $f2, gimenos1
89 s.d $f2, gnmenos2
90
91 passo2:             #calcula F(n+200) e G(n+200)
92 beq $t1, 1, condicao1 #for( int i = 1,
93 beq $t1, 2, condicao2 #for( int i = 2,
94 addi $t0, $t0, 202  #for( int i de onde parou no ultimo for
95 j loop2
96
97 condicao1:
98 li $t0, 203
99 li $t1, 3
100 j loop2
101
102 condicao2:
103 li $t0, 204
104 li $t1, 3
105 j loop2
106
107 loop2:
108 bgt $t1, $t0, saida2    # i <= (n+200); ...

```

```

109
110 #calcula F(i)
111 l.d $f2, fimenos1
112 l.d $f4, gimenos2
113 add.d $f2, $f2, $f2
114 add.d $f2, $f2, $f4
115 s.d $f2, fi
116
117 #calcula G(i)
118 l.d $f2, gimenos1
119 l.d $f4, fimenos2
120 add.d $f2, $f2, $f4
121 add.d $f2, $f2, $f4
122 add.d $f2, $f2, $f4
123 s.d $f2, gi
124
125 #ajusta para proximo passo do for
126 l.d $f2, fimenos1
127 s.d $f2, fimenos2
128 l.d $f2, fi
129 s.d $f2, fimenos1
130 l.d $f2, gimenos1
131 s.d $f2, gimenos2
132 l.d $f2, gi
133 s.d $f2, gimenos1
134
135 addi $t1, $t1, 1    # i++){
136 j loop2
137 saida2:
138
139 #calcula F(n-2) + G(n+
140 l.d $f0, fnmenos2
141 l.d $f2, gimenos1
142 add.d $f0, $f0, $f2

```


109		143	
110	#calcula F(i)	144	#(e imprime)
111	l.d \$f2, fimenos1	145	li \$v0, 4
112	l.d \$f4, gimenos2	146	la \$a0, funcao1
113	add.d \$f2, \$f2, \$f2	147	syscall
114	add.d \$f2, \$f2, \$f4	148	li \$v0, 3
115	s.d \$f2, fi	149	mov.d \$f12, \$f0
116		150	syscall
117	#calcula G(i)	151	
118	l.d \$f2, gimenos1	152	# e F(n+200) - G(n-2)
119	l.d \$f4, fimenos2	153	l.d \$f0, fimenos1
120	add.d \$f2, \$f2, \$f4	154	l.d \$f2, gimenos2
121	add.d \$f2, \$f2, \$f4	155	sub.d \$f0, \$f0, \$f2
122	add.d \$f2, \$f2, \$f4	156	
123	s.d \$f2, gi	157	#(e imprime)
124		158	li \$v0, 4
125	#ajusta para proximo passo do for	159	la \$a0, funcao2
126	l.d \$f2, fimenos1	160	syscall
127	s.d \$f2, fimenos2	161	li \$v0, 3
128	l.d \$f2, fi	162	mov.d \$f12, \$f0
129	s.d \$f2, fimenos1	163	syscall
130	l.d \$f2, gimenos1	164	
131	s.d \$f2, gimenos2	165	li \$v0, 10
132	l.d \$f2, gi	166	syscall
133	s.d \$f2, gimenos1	167	
134		168	done: #pulo pro final do programa
135	addi \$t1, \$t1, 1 # i++){	169	li \$v0, 4
136	j loop2	170	la \$a0, invalida
137	saida2:	171	syscall
138		172	
139	#calcula F(n-2) + G(n+200)	173	li \$v0, 10
140	l.d \$f0, fnmenos2	174	syscall
141	l.d \$f2, gimenos1		
142	add.d \$f0, \$f0, \$f2		