

7,5	Matrícula:	Nome:
	201910282	Igor Lima Rocha

2,0
 $x_0 = -2$

1.- (2.0 pontos)

a) Escreva um programa p-code **MONOLITICO** (sem usar rotinas)

que calcule a os valores da sequencia $x_{n+1} = x_n(1 - x_n)$, com $x_0 = -2$.

b) Simule a execução do seu programa para calcular x_2 . $x_n = (x_{n-1})(1 - x_{n-1})$

$x_2 = x_1(1 - x_1)$

X₁

$$x_1 = -2(1 - (-2)) = -6$$

$$x_2 = -6(1 - (-6)) = -42$$

2,0

2.- (2.0 pontos)

a) Escreva um programa p-code com um programa principal e uma rotina recursiva que calcule a os valores da sequencia $x_{n+1} = x_n(1 - x_n)$, com $x_0 = -2$.

b) Simule a execução do seu programa para calcular x_2 ; para cada chamada **recursiva**, na simulação, desenhe a pilha de execução, e desenhe uma seta para cada link na cadeia dinâmica.

↓ (x)
 if (x =
 rut
 auc =
 return

2,5

3.- (2.0 pontos) Considerando o (trecho de) programa p-code no verso da folha:

Identifique as instruções que correspondem ao programa principal, rotina(s) (se houverem), passagem de parâmetros por valor (se houverem), passagem de parâmetros por referencia (se houverem), mecanismo de captura de valor de retorno de função.

2,0

4.- (2.0 pontos) Considerando o (trecho de) programa p-code no verso da folha:

Faça a engenharia reversa do código, ou seja, escreva um programa em linguagem de alto nível (C-like), que corresponda às instruções p-code.

i	code[i].f	code[i].l	code[i].a
0	INT	0	50
1	LIT	0	0
2	STO	0	3
3	LIT	0	0
4	STO	0	4
5	LIT	0	0
6	STO	0	5
7	LIT	0	1
8	STO	0	6
9	LIT	0	1
10	STO	0	7
11	LIT	0	1
12	STO	0	8
13	LIT	0	0
14	STO	0	9
15	LIT	0	1
16	STO	0	10
17	LIT	0	0
18	STO	0	11
19	LIT	0	1
20	STO	0	12
21	LIT	0	0
22	STO	0	13
23	LIT	0	1
24	STO	0	14
25	LOD	0	2
26	STO	0	53 param
27	LOD	0	2
28	STO	0	54 param
29	LOD	0	3
30	STO	0	55 param
31	LOD	0	3
32	STO	0	56 param
33	LOD	0	3
34	STO	0	57 param
35	CAL	0	36
36	OPR	0	0
37	INT	0	20
38	LIT	0	0
39	STO	0	25 - param
40	LIT	0	0
41	STO	0	26 - param
42	STO	0	28 param
43	LOD	0	25
44	STO	0	53 - param
45	LOD	0	10
46	STO	0	54 - param

i	code[i].f	code[i].l	code[i].a
47	LOD	0	26
48	STO	0	55 - param
49	CAL	0	94
50	LOD	0	26
51	STO	0	10 - param
52	LOD	0	10
53	STO	0	11
54	LOD	1	8
55	LOD	1	7
56	LOD	1	6
57	LOD	1	5
58	LOD	1	4
59	LOD	1	3
60	LOD	0	3
61	LOD	0	11
62	OPR	0	3
63	JPC	0	70
64	INT	0	-1
65	LIT	0	-1
66	LOD	0	11
67	OPR	0	2
68	STO	0	11
69	JMP	0	60
70	STO	0	12
71	LOD	1	14
72	LOD	1	13
73	LOD	1	12
74	LOD	1	11
75	LOD	1	10
76	LOD	0	10
77	STO	0	11
78	LOD	1	9
79	LOD	0	6
80	LOD	0	10
81	OPR	0	3
82	JPC	0	89
83	INT	0	-1
84	LIT	0	-1
85	LOD	0	10
86	OPR	0	2
87	STO	0	10
88	JMP	0	79
89	STO	0	13
90	LOD	0	12
91	LOD	0	13
92	OPR	0	2
93	OPR	0	0

201910282

Igor Lima Rocha

X = parametro

1) 2) 0 INT 0 6

1 LIT 0 1

2 STO 0 3

3 LIT 0 -2

4 STO 0 4

5 LIT 0 X

6 STO 0 5

7 LOD 0 3

8 LOD 0 5

9 OPR 0 13

10 JPC 0 22

11 LIT 0 1

12 LOD 0 4

13 OPR 0 3

14 LOD 0 4

15 OPR 0 4

16 STO 0 4

17 LOD 0 3

18 LIT 0 1

19 OPR 0 2

20 STO 0 3

21 JMP 0 7

22 LOD 0 4

23 OPR 0 0

t	l	p	f	l	a	S
		LB	LIT	0	1, 2, 1
		19	OPR	0	2, 3
		20	STO	0	3	0, 0, 0, 3, -42, 2
		21	JMP	0	7
	7	LOD	0	3	0,	3
	8	LOD	0	5	3, 2
	9	OPR	0	13	0
	10	JPC	0	22	
	22	LOD	0	4	-42
	23	OPR	0	0		

3) programa principal: 0 → 36 ✓

rotina: 37 → 93

passagem de parametros por valor:

26, 28, 30, 32, 34, 36, 41, 42, 44, 46, 48

captura de retorno de função:

(50, 51)

parametros por referencia?

1) b) ✓	t	b	p	f	l	a	S
		0	INT	0	6	0,0,0,0,0,0	
		1	LIT	0	1	0,0,0,0,0,0,1	
		2	STO	0	3	0,0,0,1,0,0	
		3	LIT	0	-2	0,0,0,1,0,0,-2	
		4	STO	0	4	0,0,0,1,-2,0	
		5	LIT	0	2	0,0,0,1,-2,0,2	
		6	STO	0	5	0,0,0,1,-2,2	
		7	LOD	0	3	0,0,0,1,-2,2,1	
		8	LOD	0	5	0,0,0,1,-2,2,1,2	
		9	OPR	0	13	0,0,0,1,-2,2,1	
		10	JPC	0	22	0,0,0,1,-2,2	
		11	LIT	0	1	...	1
		12	LOD	0	4	...	1,-2
		13	OPR	0	3	...	3
		14	LOD	0	4	...	3,-2
		15	OPR	0	4	...	-6
		16	STO	0	4	0,0,0,1,-6,2	
		17	LOD	0	3	...	1
		18	LIT	0	1	...	1,1
		19	OPR	0	2	...	2
		20	STO	0	3	0,0,0,2,-6,2	
		21	JMP	0	7	...	
		7	LOD	0	3	...	2
		8	LOD	0	5	...	2,2
		9	OPR	0	13	...	1
		10	JPC	0	22	...	
		4	LIT	0	1	...	1
		12	LOD	0	4	...	1,-6
		13	OPR	0	3	...	7
		14	LOD	0	4	...	7,-6
		15	OPR	0	4	...	-42
		16	STO	0	4	0,0,0,2,-42,2	
		17	LOD	0	3	...	2

no
a accumulator
e stack

X = paramtro

2) a) 0 INT 0 3

1 LIT 0 X

2 STO 0 3+3

3 CAL 0 6

4 LOD 0 3+4

5 OPR 0 0

6 INT 0 6

7 LOD 0 3

8 LIT 0 0

9 OPR 0 8

10 JPC 0 14

11 LIT 0 -2

12 STO 0 4

13 OPR 0 0

14 LOD 0 3

15 LIT 0 -1

16 OPR 0 3

17 STO 0 6+3

18 CAL 0 6

19 LOD 0 6+4

20 STO 0 5

21 LIT 0 1

22 LOD 0 5

23 OPR 0 3

24 LOD 0 5

25 OPR 0 4

26 STO 0 4

27 OPR 0 0

4) int main() {

int arr = {0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1}

int x = fun(2, 2, 3, 3, 3)

return 0

}

int fun(x₁, x₂, x₃, x₄, x₅) {

fun2(0, 0, ?,

2) b)

t	b	p	f	l	a	S
0	0	0	INT	0	3	0, 0, 0
3	0	1	LIT	0	2	0, 0, 0, 2
4	0	2	STO	0	6	0, 0, 0
3	0	3	CAL	0	6	0, 0, 0, 3
9	4	6	INT	0	6	0, 0, 0, 3, 0, 3, 2, 0, 0
			LOD	0	3	0, 0, 0, 3, 0, 3, 2, 0, 0, 2
			LIT	0	0	0, 0, 0, 3, 0, 3, 2, 0, 0, 2, 0
			OPR	0	8	0, 0, 0, 3, 0, 3, 2, 0, 0, 0
			JPC	0	14	...
			LOD	0	3	..., 2
			LIT	0	1	..., 2, 1
			OPR	0	3	..., 1
			STO	0	9	...
			CAL	0	6	...

a accumulacao
nao e exatc