dez UZ, 15 20:49	0	passaro.as Page 1/17	dez 02, 15
Passaro Bamb 83449 Duarte 83539 Pedro Turno seguno 85080 Joao F	oleante Galvao Caldeira la feira,	11:00-12:30, grupo 13	; R3=Ni (Ran ; R4=Posica, ; R5= (y0) Po, ; R6= (t) Tem; ; R7=Interr; ; (T, I0, I
			Distancia
· · · · · · · · · · · · · · · · · · ·		nnan.	Nivel
INT_MASK_ADDR	E E	FFFAh	fixa (0C,0
INT_MASK	E OCI	000000000000111b	PosAPassar
BIT_MASK	D CO	100000000000000000000000000000000000000	LinhasASub
IO_DISPLAY	DOI I	FFFOh	Espobstacu.
LCD_DISP_C LCD_DISP_W	D E	FPF5가	FaseObstaci culos
TEMP_UC	EQU	FFF6h	NumObstacu
TEMP_PC	E SO	FFF7h Ooo1h	s na janel. Fetadolfo
ALLVAR_IEME LED DISPLAY	D DO	COULII FFF8h	ESCAGOLEL
TXT_DISPLAY_C	EQU	FFFCh	· ;
TXT_DISPLAY_W	E SOI	FFFEN	
DELAY COUNT		0001h	,
NIBBLE_MASK	EQU	000fh	OLNI
NUM_NIBBLES	D CO	ひゃ	THNI
NIBBLE) E	寸' ∝	LNT
COL_PASSARO	DÕE EÕO	20	INT 15
POSI_PASSARO	D G	0C00h	OTOTINT .
CH_LIMITE	D C C	1-1	
CH_OBSTACULO	EQU EQU	,×,	
CH_ZERO) (2) (3) (2) (3) (4) (4) (4) (4) (4) (4) (4) (4) (4) (4		 -
NUM_L_SUBIR	EQU	11 (2)	
DISPLAY_W	E E E O E	79	
ULTIMA_LINHA	O CO O CO O CO O CO O CO O CO O CO O CO	1700h	; En:
NUM_BOTOES	EQU	14	
LINHA GRAVIDADE		0100n 20	KInt0:
INTERVALO_OBS	EQU	5; Espaco entre cada coluna de obstaculos	
	D C		
ESPACO_OBS BASE_ DEC MAX_NIVEL	E E E	ubuun ; Espaco por onde o passaro passa 10 16	
; Palavra de me	memoria ORIG	que contem a variavel de contagem 8000h	; RIntl: Ro
Passaro	STR	, <o,< td=""><td></td></o,<>	
MsgInic11	STR	'Prepare-se' , Comprimento,	EF
MsgInicLZ MsgFimI.1	STR	'Prima o interruptor II' 'Fim do Togo'	Kinti:
MsgFimL2	STR	'0000' comprimento, 'Comprimento,	
MsgLCD	STR	'Distancia:'; Comprimento,	
			. DTato
;R1=Contador da		dificuldade	; En

;R3=Ni(Random) ;R4=Posicao do cursor ;R5=(y0)Posicao da ultima ;R6=(t)Tempo desde a ultin	cursor da uli	vez que o na vez que	passaro subiu o passaro subiu (em ciclos do temporizador)
; R'=Interrupcoe ; (T, IO, II, I	ioes ative	imeiro bit	para T e por ai em diante)
ᄶᆜᄶ	TAB TAB TAB	el el el	; Distancia percorrida ; Nivel invertido (1-16) (16-N) ; Posicao do passaro em virgula
fixa (0C,00) PosAPassaro	TAB	П	; Posicao anterior do passaro
<pre>m v.f.) LinhasASubir EspObstaculos FaseObstaculos</pre>	TAB TAB TAB	1 NUM_MAX_OBS 1	; Linhas que faltam subir ; (8043) No. maximo de obstaculos ;Fase,de 0 a 5,da movimentacao dos obsta
culos NumObstaculos s na janela	TAB	1	; Numero de obstaculos existente
EstadoLEDs	TAB	1	; Conteudo dos LEDs
· · · · · · · · · · · · · · · · · · ·			TABELA DE INTERRUPCOES
; INTO INTO INTO INTO	ORIG WORD WORD WORD ORIG	FEOON RINtO RINt1 RINt2 FEOFN RINtTemp	; Diminui dificuldade ; Aumenta dificuldade
; INICIO DO COI	CODIGO ORIG JMP	0000h Inicio	
· · · · · · · · · · · · · · · · · · ·			- INTERRUPCOES
; RIntO: Rotina da ; Entradas: ;	de R7	servico a interrupcao 7	cao do botao IO
Kinto: Eleitos	PUSH MOV ROR OR POP RTI	R1, BIT_MASK R1, 1 R7, R1	
; RIntl: Rotina de ; Entradas: ; Saidas: R?	le 27	servico a interrupcao do ?7	cao do botao Il
; Rintl: Efeitos		R1, BIT_MASK R1, 2 R7, R1 R1	
; RInt2: Rotina de ; Entradas:	a de se. as: R7	e servico a interrup R7	a interrupcao do botao 12

dez 02, 15 20:49	passaro.as Page 3/17	dez 02, 15 20:49	
S		BR. NZ	NZ DL
RInt2: Freitos: RUnt2: POSH ROW ROW OR POP	R1 R1, BIT_MASK R1, 3 R7, R1 R1	POP R POP R FET ; EscLinhaC: Rotina que e ; Entradas: (pilha)	R2 R1 que escr pilha) Li
p: Rotintradas aidas: feitos:	Ω Ω	; Saidas: ; Efeitos: EscLinhaC: PUSH PUSH	
RINCTemp: PUSH OR POP RTI	RI R1 BIT_MASK R1	MOV MOV MOV INC	
	ROTINAS AUXILIARES	MOV SUB SUB	
; ApagaEcra: Rotina	que apaga o ecra na sua totalidade 	SHL	
Saidas: Efeitos: A pagaEcra: PU	 lltera R4, Altera M[TXT_DISPLAY_C] iSH R1 PUSH R2	ELC_C1: MOV MOV MOV MOV MOV INC	
MOV	R1, CH_ESPACO R4, R0	DEC DEC BR.N	NZ
o ciclo continua AE_C1: MOV		dod dod	
MOV	M[TXT_DISPLAY_W], R1 R4	Mat a Dassas	2 2
ADD CMP CMP BR.NZ	R2, DISPLAY R4, R2 AE_C1	m •• ω	
POP POP RET	R2 R1] H H
30t	ina que desenha uma linha de um carater a escolha R4	CALL	ተኳት
Saidas: Efeitos: A esenharLim: PU PU	ltera o R4 SH R1 SH R2 ; Carater a escrever	CALL PUSH PUSH CALL	- - - - -
MOV MOV	R1, DISPLAY_W R2, CH_LIMITE	MP_C1: BR	_
o ciclo continua	; Enquanto o cursor nao chegar a ultima coluna,	; ResetTemp: Rotina : Entradas:	que
DL_L1: MOV MOV INC DEC	M[TXT_DISPLAY_C], R4 M[TXT_DISPLAY_W], R2 R4 R1	esetTem	! ! ¤
-	8		

	dez 02, 15 20:49		passaro.as Page 4/17
1		BR.NZ	DL_L1
		POP POP RET	R2 R1
	; EscLinhaC: Rotina; Entradas: (tina que s: (pilha	haC: Rotina que escreve uma linha no meio do ecra Entradas: (pilha) Linha, Mensagem Saidas:
	; Efeitos EscLinhaC:	PUSH PUSH PUSH	R1 ; Linha R2 ; Endereco da Mensagem R3 ; Comprimento da mensagem
1 1 1		MOV MOV INC MOV SUB SHR SHL	R1, M[SP+6] R2, M[SP+5] R3, M[R2] R2;Passar um a frente pois o primeiro el e o comprimento R4, DISPLAY_W R4, R3 R4, R R4, R R4, R R4, R
	ELC_C1:	MOV MOV MOV INC INC DEC BR.NZ	M[TXT_DISPLAY_C], R4 R1, M[R2] ; R1 = Letra atual M[TXT_DISPLAY_W], R1 R2 R4 R4 R3 ELC_C1
		POP POP POP RETN	R3 R2 R1 2
	; MataPassaro: Rotina ; Satidas: ; Efeitos: MataPassaro: CALL PUSH		que acaba o jogo ApagaEcra 12 MsgFimL1
		CALL PUSH CALL CALL PUSH PUSH CALL	EscLinhac M[Distancia] CalcPontuacao AtMsgFimL2 14 MsgFimL2 EscLinhaC
	MP_C1:	BR RET	MP_C1
	; ResetTemp: Rotina; Saidas:; Saidas:; Efeitos:	tina que	ativa o temporizador R1
-	_		

OW			uez uz, 13 zu.49	0.43	
MOV MOV MOV	MOV MOV MOV	R1, DELAY_COUNT M[TEMP_UC], R1 R1, AIIVAR_TEMP M[TEMP_PC], R1		MOV SUB INC	
POP	T.E	R1		; Ver SUB BR.NN	Verific B R NN C
; TelaDeJogo: Roti ; Entradas: ; Saidas:	ina qu 	otina que desenha os limites s:		, Se MOV BR	e R1
TelabeJogo: MOV CALI MOV CALI RET	MOV CALL MOV CALL RET	R4, R0 DesenharLim R4, ULTIMA_LINHA DesenharLim	CP_1:	MOV INC DIV INC	Se R1
			CP_2:	VOM	Devolve N
	ualizar a M S: Pilha - p M[MSgFimL2]	r a MsgFimL2 conforme a pontuacao a - pontuacao imL2]		POP POP RET	
AtMsgFimL2: Pu Pu Pu Pu	PUSH	R1 ; Pontuacao R2 ; Endereco a escrever R3 ; Algarismo a escrever	; EscDistancia: ; Entradas: ; Saidas:		in [D]
MOV MOV ADD ;Enquanto nao		R1, M[SP+5] R2, MsgFimL2 R2, M[R2] forem escritos na memoria todos os elementos,o ciclo conti	; EscDistancia:	breitos: Aitera R ncia: PUSH R PUSH R PUSH R) -
AMFL2_C1: MOV DIV	20	()		CALL	ы
MOV DEC CMP BR.	MOV DEC CMP BR.NZ	ADD R3, CH_ZERO M[R2], R3 R2 R1, R0 AMFL2_C1	ED_C1:	/ Descobr MOV R MOV R INC R	O O
POP POP POP RETI	POP POP POP RETN	R3 R2 R1		BR.NZ E BR.SZ ; ESCreve MOOV	NZ SC
CalcPontuacao: Entradas Saidas: Efeitos	Rotina S: Pilha Pilha	que - di pont	ED_C2:	MOV MOV MOV MOV	
CalcPontuacao: PU PU	PUSH R1 PUSH R2	Rl ; Distancia R2 ; DISPLAY_W - COL_PASSARO + 1		MOV DEC CMP	2
. • .	(1) 1	Pontuacao = 0, se R1 < R2 Pontuacao = (R1 - R2)/(INTERVALO_OBS + 1) + 1, se R1 >= R2		404 404	!
II ;	Inic.	Inicializar R1 V R1, M[SP+4]		POP POP	
`	Inic.	Inicializar R2		KET	

	dez 02, 15 20:49		passaro.as Pag	Page 6/17
		MOV SUB INC	R2, DISPLAY_W R2, COL_PASSARO R2	
		; Verif SUB BR.NN	SUB R1, R2 BR.NN CP_{-1}	
		; Se R1 MOV BR	< <i>R2, executar (1)</i> R1, R0 CP_2	
	CP_1:	; Se R1 MOV INC DIV INC	>= R2, executar (2) R2, INTERVALO_OBS R2 R1, R2 R1, R2	
1.1	CP_2:	; Devol	; Devolver pontuacao MOV M[SP+4], Rl	
		POP POP RET	R2 R1	
	; EscDistancia: Rotina ; Entradas: M[Dis	tancia: Rotina que esc Entradas: M[Distancia]	que escreve a distancia percorrida no display LCD.	
Ţ,	; Saldas: ; Efeitos EscDistancia:	: Altera PUSH PUSH PUSH PUSH	Saldas: Efeitos: Altera M[LCD_DISP_C] ncia: PUSH R1; Coordenada do LCD PUSH R2; Distancia PUSH R3; Algarismo a ser escrito PUSH R4; Numero de algarismos da distancia	
		CALL	EscLCD	
	ED_C1:	; Descobrir MOV R4, INC R4 MOV R3, DIV R2,	brir o numero de algarismos da distancia R2, M[Distancia] R4, R0 R4 R3, BASE_ DEC R2, R3 ED_C1	
	ED_C2:	; Escre MOV ADD MOV DIV ADD MOV MOV DEC CMP BR.NZ	Escrever a distancia no LCD NY R2, M[Distancia] DD R1, R4 NY R3, BASE_DEC NY R2, R3 NY M[LCD_DISP_C], R1 NY M[LCD_DISP_W], R3 CR R1 R2, R0 R2, R0 R2, R0 R3, CR R1 R2, R0 R3, R4 R4, R6 R5, R6	
2		POP POP POP RET	R4 R3 R2 R1	
_				

0.01			
; EscIO: Rotina q ; Saidas: - ; Saidas: - ; EscIO: Ffeitos: PPI	gue Posh Posh Posh Posh	efectua a escrita da pontuacao RI ; Distancia RS R3 R4	; ApagaPassaro; ; Saidas; ; Saidas; ; Efeito: ApagaPassaro:
	MOV MOV PUSH CALL POP	R2, NUM_NIBBLES R3, IO_DISPLAY M[Distancia] CalcPontuacao R1	
EIO_1:	MOV DIV MOV INC DEC BR.NZ	R4, BASE_DEC R1, R4 M[R3], R4 R3 R2 EIO_1	AP_C1: screver
	POP POP POP RET	R4 R3 R2 R1	
; EscLCD: Rotina; ; Entradas: ; Saidas: R	na que es 15: R1: coc	EscLCD: Rotina que escreve 'Distancia:' no display LCD: Entradas: Santagas: Efficas: R1: coordenada atual do LCD	; DesPassaro:
4 <i>rotin</i> scLCD:	rre cada PUSH PUSH PUSH	letra da mensagem definida e vai inseri-la no lcd R2 ; Endereco da mensagem R3 ; Comprimento da mensagem R4 ; Letra a ser escrita	; Saidas; ; Efeitos DesPassaro:
	; Limpar MOV BOR BOD BOOV BOOV BOOV BOOV BOOV BOOV BOOV	RI, BIT_MASK RI, 15 e 5) RI, BIT_MASK RI, 10 RI, BIT_MASK MILCD_DISP_C], RI	
	MOV	R1, BIT_MASK ; Primeiro bit a 1, para ligar o display	
ELCD_C1:	MOV MOV INC MOV MOV INC DEC BR.NZ	R2, MsgLCD R3, M[R2] M[LCD_DISP_C], R1 R4 M[LCD_DISP_M], R4 R1 R1 R1 R1 R3 ELCD_C1	DP_C1:
	POP POP RET	R4 R3 R2	

; ApagaPassaro: Rotina que apaga; Butadas: MIPosAPassaro]; Saidas:		
S.	: Kotina as: M[Pos. :	que apaga a posicao anterior do passaro sAPassaroj
	s: Altera PUSH PUSH	a R4, M[TXT_DISPLAY_C], M[TXT_DISPLAY_W] R1 R2
	MOV R1 MOV R2 ; Calcular MOV R4	R1, TAM PASSARO R2, CH_ESPACO lar a posicao anterior do passaro R4, M[PosApassaro]
	MVBL	RO 'PORTOS a coluna a O para depois adicion
AP_C1:	MOV	N4, COL_FASSANO M[TXT_DISPLAY_C], R4 ; Apontar o cursor para a posicao do passaro e e
screver	MOV INC DEC BR.NZ	MOV M[TXT_DISPLAY_W], R2 INC R4 DEC R1 BR.NZ AP_C1 FERGUANTO nao eliminar o passaro repete o ciclo
	POP	POP R2
; DesPassaro: Rot ; Entradas:	otina 3:	que desenha o passaro
Efficación Espassaro:		a R4 R1 ; Endereco do passaro R2 ; Carater a ser escrito R3 ; Tamanho do passaroR4 ; Coordenada do passaro
	PUSH R5 Calcular MOV R4 MVBL R4 ADD R4 MOV R1 MOV R1	R5 R4, M[PosPassaro] R4, R0 R4, R0 R4, COL_PASSARO R1, Passaro R3, TAM_PASSARO
DP_C1:	MOV R2, 'Verificar PUSH R4 CALL COO. POP R5, CMP R5,	M[R1] se estamos a o rdTemobs R0 aPassaro
	MOV MOV INC INC DEC BR.NZ	M[TXT_DISPLAY_C], R4 M[TXT_DISPLAY_W], R2 R4 R1 R3 DP_C1
	POP	R5 R3

dez 02, 15, 20:49		Dassaro.as	dez 02, 15,20%
	POP POP RET		
<pre> ; DescerPassaro: Rotina que de. ; Entradas: M[PosPassaro]; ; Saidas: M[PosPassaro], ; Efeitos: DescerPassaro: PUSH R1 PUSH R2</pre>	Rotina S: M[PosPa M[PosPa PUSH	Passaro: Rotina que desce o passaro conforme a acao da gravidade Entradas: M[PosPassaro], M[PosAPassaro], R5, R6 Saidas: M[PosPassaro], M[PosAPassaro] Efeitos: ssaro: PUSH R1 PUSH R2	subiu SP_Fim:
	MOV MOV SUB MOV CMP ; y = y ; v = y ; v = y ; v = y ; v = y MOV MOV MUL	R1, M[PosPassaro] R2, ULTIMA_LINHA R2, LINHA M[PosAPassaro], R1 R1, R2 , R1 > R2 (ultima linha) , MataPassaro y0 + v0t + (1/2)*g*t^2 = 0, y0 = R5, t = R6 g = GRAVIDADE = GRAVIDADE * R6 * R6 / 2 + R5 R1, GRAVIDADE R2, R6 R2, R6 R2, R1 R2, R1 R2, R1 R2, R3	; VPosPassaro; ; Saida ; ; Efeit VPosPassaro:
	MUL SHR ADD MOV POP		VPP_DescPass:
	POP RET	R1	VPP_Fim:
; SubirPassaro: R; Entradas: ; Saidas: . ; Efeitos: SubirPassaro: Pta o passaro Pbir	Rotina : PUSH	que sobe o passaro "Linha em que es R1 "Num linhas a su	; DesX: Rotin ; DesX: Rotin ; Saida ; Saida ; Efeit
A W W W W	12333	R6, R0 R1, M[PosPassaro] M[PosAPassaro], R1; Guardar a posicao anterior do passaro R2, NUM_L_SUBIR R2, BYTE; Converter 000Xh em 0X00h estava na parte das	
	BE BE S.N. N.	R1, R2 R1, R2 Subir o passaro NUM_L_SUBIR linhas R1, M[LinhasASubir] R1, LIM_TOPO ; Se Pos-NUM_L_SUBIR < 0100h: Pos = 0100h R5, R1 R5, R1	
ubiu	SUB ADD MOV SUB	·H ()	; ApagaUltCol; Entra ; Saida ; Efeit ApagaUltCol:
CO ordered crief chair			

R1, LIM_TOPO R5, R1 R1, LIM_TOPO R6, R1 R1 R1 R1 R1 M[LinhasAsubir], R0 WPP_Descreases R1, M[PosPassaro], R1 M[LinhasAsubir], R0 WPP_Descreases R1, LINHA M[PosPassaro], R1 M[LinhasAsubir], R1 M[LinhasAsubir], R1 M[LinhasAsubir], R1 M[LinhasAsubir], R1 R6 DescerPassaro] R1 R1 R1 R4 M[TXT_DISPLAY_C], R4 R1, CH_ESPACO M[TXT_DISPLAY_C], R4 R1, CH_ESPACO M[TXT_DISPLAY_C], R4 R1, CH_ESPACO M[TXT_DISPLAY_M], R1 R4 R1, CH_OBSTROULO M[TXT_DISPLAY_M], R1 R4 R1, CH_OBSTROULO M[TXT_DISPLAY_M], R1 R1 R1 R1 R1 R1 R2 R1 R2 R1 R1	dez 02, 15 20:49	<u>ရ</u>	passaro.as Page 10/17
MOV R1, LIM_TOPO MOV R5, R1 ; guardar a posicao da ultima v POP R2 POP R1 FET cos		BR	SP_Fim
POP R2 POP R1 POP R2 POP R1 RET POP R1 RET POP R1 POP R1 POP R2 POP R1 POST R2 POST R1 MOV M[LinhasASubir], R0 BR R1 LINHA SUB M[LinhasASubir], R1 BR M[LinhasASubir], R1 RET POP R1 NOV M[TXT_DISPLAY_C], R4 NOV M[TXT_DISPLAY_C], R4 NOV M[TXT_DISPLAY_M], R1 POP M[TXT_DISPLAY_M], R1 POP M[TXT_DISPLAY_M], R1 RET RET RET ROTING que apaga a ultima coluna da esquerda da cdas:	SP_1:	MOV	LIM_TOPO R1 ; guardar a posicao da
POP R2 RET CALL POP R1 SUB MOV MILIANDASASUDIS], R0 MOV MILIANDASASUDIS], R1 MOV MILIANDASASUDIS], R1 MOV MILIANDASASUDIS], R1 MOV MILIANDASASUDIS], R1 SUB MILIANDASASUDIS], R1 SUB MILIANDASASUDIS], R1 RET INC R6 CALL DescerPassaro MOV MIXT_DISPLAY_C], R4	subiu	MOV	
core not not not continue as one passaro tem linhas a sun cidas: bush RI cMP MilinhasAsubir], R0 BR.Z VPP_DescPass MOV R1, LINHA SUB MilosAbasaro], R1 SUB MilohasAsubir], R1 BR WPP_Fim INC R6 CALL DescerPassaro RET RET RET RET ROP R1 RET ROP R1 RET ROP R1	SP_Fim:	POP POP RET	R2 R1
CNUP BR.Z WPP_DescPass MOV RI, M[PosPassaro] MOV RI, M[PosPassaro], R1 SUB M[LinhasASubir], R1 BR VPP_Fim INC R6 CALL DescerPassaro POP R1 R2 R2 R2 R4 COordenada do ecra cos: PUSH NOV M[TXT_DISPLAY_C], R4 MOV M[TXT_DISPLAY_W], R1 DEC C Desenha um X MOV M[TXT_DISPLAY_W], R1 DEC C MOV MITXT_DISPLAY_W], R1 DOP R1 C Coordenada atual PUSH R2 C COORDENADA atual PUSH	; VPosPassaro;; Saida: ; Saida: ; Efeit(α •• α	se o passaro tem linhas a
TNC R6 CALL DescerPassaro POP R1 SET das: R4 - Coordenada do ecra idas: R4 - Coordenada do ecra push NOV R1, CH_ESPACO MOV R1, CH_OBSTACULO MOV R1, CH_O		CMP BR.Z MOV MOV MOV SUB SUB BR	M[LinhasASubir], R0 VPP_Descrass R1, M[PosPassaro] M[PosPassaro], R1 R1, LINHA M[PosPassaro], R1 M[LinhasASubir], R1 VPP_Fim
POP RET	VPP_DescPass:	INC	R6 DescerPassaro
a que desenha um X e apaga o carater a direita. isa que desenha um X e apaga o carater a direita. is: is que desenha um X e apaga o carater a direita. is:	VPP_Fim:	POP RET	R1
a que desenha um X e apaga o carater a direita. idas: R4 - Coordenada do ecra is: push R1 INC R4 MOV M[TXT_DISPLAY_C], R4 MOV M[TXT_DISPLAY_M], R1 DEC R4			
MOV M[TXT_DISPLAY_C], R4 MOV R1, CH_ESPACO MOV R1, CH_ESPACO MOV M[TXT_DISPLAY_W], R1 > Desenha um X MOV M[TXT_DISPLAY_C], R4 MOV R1, CH_OBSTACULO MOV R1, CH_OBSTACULO MOV R1 FET **Rotina que apaga a ultima coluna da esquerda da cidas: is: is: ios: Altera R4, M[TXT_DISPLAY_W] **PUSH** R1, Coordenada atual **PUSH** R2, inha inha inha	 ×	le l	um X e apaga o carater a denada do ecra
RET RET: Rotina que apaga a ultima coluna da esquerda da sis: IS: OS: Altera R4, M[TXT_DISPLAY_C], M[TXT_DISPLAY_W] PUSH PUSH R2: Iltima linha		INC MOV MOV MOV DEC ; Dese MOV MOV	DISPLAY_C], _ESPACO DISPLAY_W], DISPLAY_C], OBSTACULO DISPLAY_C],
: Rotina que apaga a ultima coluna da esquerda da ess is: os: Altera R4, M[TXT_DISPLAY_C], M[TXT_DISPLAY_W] PUSH R1, Coordenada atual PUSH R2: Illina linha		POP RET	R1
1/2 / O1 C11111	; ApagaUltCol ; Entrac ; Saida: ; Efeit. ApagaUltCol:	Rotina as: s: Alter PUSH	ue apaga a ultima coluna da esquerda R4, M[TXT_DISPLAY_C], M[TXT_DISPLAY_ R1 ; Coordenada atual R2 ; Ultima linha

dez 02, 15 20:49		passaro.as Page 11/17	dez 02, 15 20
	PUSH		DO_Fim:
AUC_C1:	MOV MOV MOV MOV MOV ADD CMP BR.NZ	RI, LINHA R2, ULTIMA_LINHA R3, CH_ESPACO M[TXI_DISPLAY_C], R1 M[TXI_DISPLAY_W], R3 R1, LINHA R1, R2 AUC_C1; Enquanto a linha nao for a ultima linha	; DeslocaObs
	POP POP RET	R3 R2 R1	; ; DeslocaObs:
; DesObs: Rotina ; Entradas: ; Saidas: - ; Efeitos:	gue	desenha os obstaculos RI : Numero de obstaculos	for a ultim
	PUSH PUSH FUSH PUSH	RI, Numero de observatos R3; Ultima linha para desenhar obs R4: Posicao do cursor PUSH R5; (registo auxiliar) R6; Endereco do vetor aleatorio R7; Numero aletorio	DO_1: DO_2:
	CMP JMP.Z CALL MOV SUB MOV MOV DEC SUB MOV	Mov R1, M[Numobstaculos] R1, R0 ;se o numero de obstaculos for 0,a rotina que DO_Fim ; desenha os obstaculos e passada a frente ApagaUltCol R3, ULTIMA_LINHA R6, Espobstaculos R4, DISPLAY_W R4, M[FaseObstaculos] R7, M[R6]	; IncNumObs:
DO_C1:	ADD MOV SHR CMP BR.NZ	Z	INO_Skip: ; InsereVetO
DO_NaoFazerEsp:		DesX R4, R3 ; Cursor <ultima linha<br="">DO_C1 R6</ultima>	; ; InsereVetObs
_	MOV MVBH SUB DEC BR.N	R7, M[R6] R4, R0 R4, INTERVALO_OBS R4 DO_Fim ; Se a coluna for negativa (fora do ecra	r de Ni
	DEC BR.NZ	R1 D0_C1	

dez 02, 15 20:49		passaro.as Page 12/17
DO_Fim:	POP POP POP POP POP RET	R7 R6 R5 R3 R1 R1
; ; ; Deslocabbs: Deslocabs:	Rotina que Entradas: Saidas: - Efeitos: PUSH R	<pre>que desloca todos os obstaculos para a esquerda idas: is: ios: R1 ; Fase dos obstaculos R2</pre>
	MOV CMP BR.NZ	R1, M[FaseObstaculos] R1, INTERVALO_OBS DO_1 ; (Se a fase nao
for a uitima)	MOV CALL CALL BR	M[FaseObstaculos], R0 InsereVetObs IncNumObs DO_2
DO_1:	INC INC CALL CALL	M[FaseObstaculos] M[Distancia] EscDistancia
	POP POP RET	R2 R1
; IncNumObs:Rotina; En; Sa; Sa, Eff IncNumObs: PU	ina que Entrada: Saidas: Efeitos PUSH	na que incrementa o numero de obstaculos se nao atingiu o maximo Entradas: M[NumObstaculos] Saidas: M[NumObstaculos] Efeitos:
	MOV CMP BR.Z INC	R1, NUM_MAX_OBS M[NumObstaculos], R1 INO_Skip M[NumObstaculos]
INO_Skip:	POP	R1
; insereVetObs:	ina rada das: itos	<pre>que insere um novo elemento no vetor dos obstaculos is: : R1 R2</pre>
7	CALL CALL MOV	MemObsFrente RandomN R1, R3 ; R1 toma o valo
r de Ni	; 2 <: ; 0 <= 1 MOV SUB SHR	<pre><= N_ALEATORIO <= ULTIMA_LINHA - ESPACO_OBS - 2 N_ALEATORIO - 2 <= ULTIMA_LINHA - ESPACO_OBS - 4 R2, ULTIMA_LINHA R2, ESPACO_OBS R2, BYTE</pre>

dez 02, 15 20:49		passaro.as Page 13/17	dez 02, 15 20:49
	SUB DIV ; N_ALLE ADD MOV	<pre>SUB</pre>	; Atraves de cai ; passaro, ele e
	POP POP RET	R2 R1	; Se houver mend
<pre> ; MemObsFrente:Rotina que desloca ;</pre>	Rotina que Entradas: Saidas: - Efeitos: PUSH R PUSH R	<pre>uve desloca os enderecos de memoria dos obs para a direita R1 ; Indice R2 ; Endereco do objeto</pre> R3	; Coordenada da
	MOV SUB MOV ADD	RI, NUM_MAX_OBS RI, 2 R2, Espobstaculos R2, R1	
		; Equivale A "MOV M[R2+1], M[R2]" ; passamos o R2 que esta a frente para tras	
MOF_C1:	MOV INC MOV SUB DEC BR.NN	R3, M[R2] R2 M[R2], R3 R2, 2 R1 MOF_C1	CTO_C1: ,1
	POP POP POP RET	R3 R2 R1	
; RandomN: Rotina que ; Entradas: ; Saidas: ; Efeitos: RandomN: BR.NZ ROR RET		gera um numero aparentemente aleatorio R3, 0001h RN_2 R3, 1	CTO_False: CTO_True:
RN_2:	XOR ROR RET	R3, 1 R3, 1	CTO_Fim:
		COLISOES	
Cooorc	Rotina s: pilh pilha	que verifica se a coordenada dada e um obstaculo a - coordenada - boolean: e obstaculo	; DecNivel: Rots; Entradas
; Efeitos: CoordTemObs:	PUSH PUSH PUSH PUSH		; Saidas: ; Efeitos: DecNivel:
word diet of one	C CO oxdmozob	2007	

	07.00 15 20.40		25 Oresean
	2	PUSH	TECONOMICS OF THE TOTAL OF THE
	; Atraves de ca ; passaro, ele	დ დ	lia co
d d	; Se houver men) T Z	0 obstaculos no ecra, nao pode ser obstaculo. R3, M[NumObstaculos] R3, 10 CTO_False R1, M[SP+7]; coordenada a ser verificada
	; Coordenada da ; Coordenada da ; = - 2^	MOV SHR MVBH MOV COL D: COL 9: 3 * INTE MOV SHL	R2, R1 R2, R1 R1, BYTE ; isolar a linha R2, R0 ; isolar a coluna R5, EspObstaculos n: DISPLAY_W - 1 - Fase - n * (INTERVALO_OBS + 1) 9: DISPLAY_W - 10 - Fase - 9 * INTERVALO_OBS INTERVALO_OBS - INTERVALO_OBS R4, INTERVALO_OBS R4, 3
		ADD NEG ADD SUB SUB	
	CTO_C1:	CMP BR.Z ;Nao esta	R2, R4; coordenada esta numa coluna com obstaculos? CTO_1 a numa coluna com obstaculos,logo nao pode ser obstaculo.
	CTO_1:	BR ; Coorc ADD MOV ; Para	CTO_F enada RS, 9 RS, N
		SUB BR.P MOV SHR ADD BR.NP	R5, R1 CTO_True R4, ESPACO_OBS R4, BYTE R5, R4 CTO_True
	CTO_False:	MOV	R1, R0 CTO_Fim
	CTO_True:	MOV	R1, 1
	CTO_Fim:	MOV	M[SP+7], R1 R5
		POP	POP R4 POP R3 POP R2 R1 R2
	; DecNivel: Rotina; ; Entradas:	tina que as:	diminui a dificuldade do jogo
	ecNivel	HSD4	R2
_			

dez 02, 15 20:49		passaro.as Page 15/17	dez 02, 15 20:49
	MOV CMP BR.Z INC SHL MOV MOV	R2, MAX_NIVEL M[Nivel], R2 DecNivel_Skip Mivel] M[StadoLEDS], 1 R2, M[EstadoLEDS] M[LED_DISPLAY], R2	
DecNivel_Skip:	POP RET	R2	
; IncNivel: Rotina que ; Entradas: ; Saidas: ; Efeitos: IncNivel: PUSH	ina que 1S: ::	aumenta a dificuldade do jogo R2	
	MOV CMP BR.Z DEC SHRA MOV MOV MOV MOV DEC CMP BR.NP	R2, 1 M[Nivel], R2 IncNivel_Skip M[Nivel] M[Nivel] M[StadolEDS], 1 R2, M[EstadolEDS] M[LED_DISPLAY], R2 Odiminuir o N, verificar que o contador nao ultrapassa N-1 R2, M[Nivel] R2, R2 = N - 1 R1, R2; Contador <= N - 1 R1, R2; Contador <= N - 1 R1, R2; Contador Skip R1, R2; Se ultrapassar, Contador = N - 1	
IncNivel_Skip:	POP RET	R2	
; Tempobstaculos:Rotina;; Entradas: ; Saidas: ; Efeitos: Tempobstaculos: PUSH	s:Rotina s: :: PUSH	que trata da temporizacao dos obs conforme o nivel R2	Esperar_Inic:
,	MOV DEC CMP BR.NZ ; Se CO MOV CALL	R2, M[Nivel] R2 R1, R2 T0_SkipObs contador estiver no maximo R1, R0 ; Reset ao contador DeslocaObs ; Execucao do deslocamento dos obstaculo	
'n	CALL BR ; Caso	Desobs TO_Fim contrario	
TO_Skipobs: To_Fim:	INC POP RET	R1 R2	Ciclo_1:
		INICIO	ciclo global
;	programa MOV MOV MOV	R1, SP_INICIAL SP, R1 R1, INT_MASK	

dez 02, 15 20:49	49	passaro.as	Page 16/17
	MOV	M[INT_MASK_ADDR], R1	
	, Inic. MOV	<pre>Inicializar variaveis do jogo</pre>	
	; Inic. MOV MOV CALL MOV MOV PUSH PUSH CALL PUSH CALL PUSH CALL PUSH CALL PUSH CALL PUSH CALL PUSH PUSH PUSH PUSH ROV MOV MOV MOV SHR ENI	Inicializar porto de controlo da janela de texto V R4, LIMPAR_JANELA V M[TXT_DISPLAY_C], R4 LL R5 R9 V R3, R0 V R3, R0 SH MsgInicl1 LL EscLinhaC INicializar variaveis do passaro V R6, R0 V R6, R0 V R7, R1 V R7, R1 IL EscLinhaC INicializar Variaveis do Passaro V R6, R0 V R7, R1 V R6, R1 V R7, R2, Z	
isperar_Inic:	S C C C C C C C C C C C C C C C C C C C	ist R7, R2 Esperar_Inic B R7, R2 LB R7, R2 LB R9daEcra Ativar temporizador N INT_MASK_ADDR], R1 Desenhar limites do jogo LL ResetTemp N R1, R0 N R1, R0 N R1, R0 N R2, M[EstadoleDs] MOV R2, MLED_DISPLAY], R2	
	ENI MOV DSI	R2, BIT_MASK ; As interrupcoes sao desligadas dur	durante todo o
sicio giobai	; para CMP ; sign.	 para assegurar a ordem correta das operacoes. P M[PosAPassaro], R0 ; Se a posicao anterior for diferente significa que houve uma subida do passaro, ; logo tera que desenha-lo de novo. 	de zero,