

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 1/18
; Projeto Arquitectura de Computadores
; Joao Figueiredo N75741
; Duarte Goncalves N66251
;
; ZONA I: Constantes
INT_MASK_ADDR EQU FFFAh ; Endereço da Mascara de Interrupcoes
INT_MASK EQU 1001101110011111b ; Mascara de Interrupcoes
SP_INICIAL EQU FFFFh ; Pilha
IO_Control EQU FFFCh ; Controlo da Janela de Texto
LCD_Control EQU FFF4h ; Controlo do LCD
LCD_WRITE EQU fff5h ; Escrita no LCD
TimerValue EQU FFF6h ; Endereco do conteudo do temporizador
TimerControl EQU FFF7h ; Endereco do controlo do temporizador
LEDControl EQU FFF8h ; Controlo dos LEDs
TimeLong EQU 0001h ; Conteudo do temporizador
EnableTimer EQU 0001h ; Controlo do temporizador
Segmento71 EQU FFF0h ; Endereco do primeiro display 7
segmentos da direita
Segmento72 EQU FFF1h ; Endereco do segundo display 7
segmentos da direita
FIM_TEXTO EQU '@' ; Caracter que identifica o fim de uma string
SimboloRobot1 EQU '^'
SimboloRobot2 EQU '<'
SimboloRobot3 EQU '>'
SimboloChao EQU '-'
SimboloEspaco EQU ' '
SimboloMissil EQU '|'
SimboloMeteoro EQU '*'
RANDOM EQU 1000000000010110b ; Mascara para o algoritmo para gerar numeros aleatorios

; ZONA II: Variaveis
ORIG 8000h
LASTRANDOM WORD 0 ; Ultimo numero aleatorio gerado
EstadoJogo WORD 0 ; Estado do jogo
; Estado do Jogo: 0 - ecrã inicial, 3 - definicoes, 4 - instrucoes, 1 - jogo, 2 - fim de jogo
Score WORD 0
MaxScore WORD 0
LastScore WORD 0
UserPoints WORD 0
PosicaoRobot WORD 5672
FlagMoveCanhao WORD 2
FlagME WORD 0
FlagMD WORD 0
FlagMissilReady WORD 1 ; Permite lancar o missil ou nao
FlagLancaMissil WORD 0 ; Ativa quando o jogador prime I
2
TimeGeraMeteoro WORD 20
ContGeraMeteoro WORD 20
ContadorMissil WORD 0
MissilPosicao WORD 0
MissilContador WORD 0
VarTexto1 STR 'Bem-vindo a chuva de Meteoros!', FIM_TEXTO
VarTexto2 STR 'Prima o interruptor I1 para comecar', FIM_TEXTO
VarTexto3 STR 'Prima o interruptor I2 para ver as instrucoes', FIM_TEXTO
VarTexto4 STR 'Prima o interruptor I3 para alterar definicoes', FIM_TE

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 2/18
XTO
VarDefinicoes STR 'Definicoes', FIM_TEXTO
VarDefinicoes1 STR 'I7 - Quantidade de Meteoros no ecrã: ', FIM_TEXTO
VarDefinicoes2 STR 'I9 - Velocidade de Meteoros: ', FIM_TEXTO
VarDefinicoes3 STR 'Varios', FIM_TEXTO
VarDefinicoes4 STR 'Um ', FIM_TEXTO
VarDefinicoes5 STR 'Estavel ', FIM_TEXTO
VarDefinicoes6 STR 'Aleatoria', FIM_TEXTO
VarDefinicoes7 STR 'I8 - Velocidade para gerar meteoro: ', FIM_TEXTO
VarDefinicoes8 STR 'Aleatoria', FIM_TEXTO
VarDefinicoes9 STR 'Fixa ', FIM_TEXTO
VarInstrucoes STR 'Instrucoes', FIM_TEXTO
VarInstrucoes1 STR 'O objetivo do jogo e impedir os meteoros de chegarem a Terra.', FIM_TEXTO
VarInstrucoes2 STR 'Para mover o canhao utiliza os interruptores IO e IB.', FIM_TEXTO
VarInstrucoes3 STR 'Para lançar o missil utiliza o interruptor I2', FIM_TEXTO
VarInstrucoes4 STR 'Poderas lançar um missil sempre que os 16 LEDs estiver em acessos', FIM_TEXTO
VarInstrucoes5 STR 'O jogo termina quando tres meteoros chegarem a Terra.', FIM_TEXTO
VarPontos STR 'Pontos: 0000 *0', FIM_TEXTO
VarMaximo STR 'Maximo: 0000', FIM_TEXTO
VarFimJogo1 STR 'Fim do Jogo', FIM_TEXTO
VarFimJogo2 STR 'Prima o interruptor I1 para recomecar', FIM_TEXTO
EspacoLimpaEcrã STR ' ', FIM_TEXTO

DefMeteorosQt WORD 1
DefMeteorosVel WORD 1
DefMeteorosCont WORD 1
ContMoveMeteoro WORD 0
QtMeteoros WORD 0
MeteoroPos WORD 0
MeteoroCont WORD 0
MeteoroVel WORD 0

; ZONA III: Tabela de Interrupcoes
ORIG FE00h
INT0 WORD Interruptor0
INT1 WORD Interruptor1
INT2 WORD Interruptor2
INT3 WORD Interruptor3
ORIG FE07h
INT7 WORD Interruptor7
INT8 WORD Interruptor8
INT9 WORD Interruptor9
ORIG FE0Bh
INTB WORD InterruptorB
ORIG FE0Fh
INTTemp WORD Temporizador

; ZONA IV: Codigo
ORIG 0000h
MOV R7, SP_INICIAL ; Inicia a pilha
MOV SP, R7 ;
MOV R7, INT_MASK ;
MOV M[INT_MASK_ADDR], R7 ; Ativa a mascara de interrupcoes
s
MOV R7, R0 ; Limpa R7

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 3/18
MOV M[EstadoJogo],R0 ; Coloca o Estado do Jogo como I
nicial (Ecra Inicial)
JMP Inicio

;INT0
; Entradas: ---
; Saidas: ---
; Efeitos: Altera FlagME
Interruptor0: PUSH R1
MOV M[FlagMoveCanhao],R0 ; Flag MoveCanhao=0
POP R1
RTI

;INTB
; Entradas: ---
; Saidas: ---
; Efeitos: Altera FlagMD
InterruptorB: PUSH R1
MOV M[FlagMoveCanhao],R0 ; Flag MoveCanhao=1
INC M[FlagMoveCanhao]
POP R1
RTI

;INT1
; Entradas: ---
; Saidas: ---
; Efeitos: Altera EstadoJogo
Interruptor1: PUSH R1
MOV R1,3
CMP M[EstadoJogo],R1 ; Caso o estado do jogo sejam as
definicoes volta-se ao ecra inicial
BR.NN VoltaMenu
MOV R1,2
CMP M[EstadoJogo],R1 ; Caso o estado do jogo seja fim
de jogo, recomeca-se o jogo
BR.Z RecomecaJogo
CMP M[EstadoJogo],R0 ; Caso o estado do jogo seja o e
cra inicial, começa o jogo
BR.Z RecomecaJogo
BR FimJogoFlag
VoltaMenu: MOV M[EstadoJogo],R0
BR FimJogoFlag
RecomecaJogo: MOV R1,1
MOV M[EstadoJogo],R1
FimJogoFlag: POP R1
RTI

;INT2
; Entradas: ---
; Saidas: ---
; Efeitos: Altera EstadoJogo ou Lanca Missil
Interruptor2: PUSH R1
MOV R1,1
CMP M[EstadoJogo],R1 ; Caso o estado do jogo seja a j
ogar, ativa flag LancaMissil
BR.Z SaltoINT2_3
SaltoINT2: CMP M[EstadoJogo],R0 ; Caso seja ecra inicial, aprese
nta as instrucoes
BR.NZ FimInstrucoes
MOV R1,4
MOV M[EstadoJogo],R1
SaltoINT2_3: CMP M[FlagMissilReady],R0 ; Caso a flag MissilReady esta a

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 4/18
tiva, ativa flag LancaMissil
BR.Z FimInstrucoes ; Caso contrario termina rotina
MOV M[FlagLancaMissil],R1
MOV M[FlagMissilReady],R0
MOV M[ContadorMissil],R0
FimInstrucoes: POP R1
RTI

;INT3
; Entradas: ---
; Saidas: ---
; Efeitos: Altera EstadoJogo
Interruptor3: PUSH R1
CMP M[EstadoJogo],R0 ; Caso o estado do jogo seja ecr
a inicial, apresenta as definicoes
BR.NZ FimDefinicoes
MOV R1,3
MOV M[EstadoJogo],R1
FimDefinicoes: POP R1
RTI

;INT7
; Entradas: ---
; Saidas: ---
; Efeitos: Altera definicoes
Interruptor7: PUSH R1
MOV R1,3
CMP M[EstadoJogo],R1 ; Caso o estado do jogo seja def
inicoes, altera as definicoes associadas ao I7
BR.NZ FimInterruptor7
CMP M[DefMeteorosQt],R0
BR.Z Interruptor71
MOV M[DefMeteorosQt],R0
MOV R2,VarDefinicoes4
MOV R1,110000101111b
CALL EscString ; Escreve alteracao no ecra
BR FimInterruptor7
Interruptor71: INC M[DefMeteorosQt]
MOV R2,VarDefinicoes3
MOV R1,110000101111b
CALL EscString ; Escreve alteracao no ecra
FimInterruptor7: POP R1
RTI

;INT8
; Entradas: ---
; Saidas: ---
; Efeitos: Altera definicoes
Interruptor8: PUSH R1
MOV R1,3
CMP M[EstadoJogo],R1 ; Caso o estado do jogo seja def
inicoes, altera as definicoes associadas ao I8
BR.NZ FimInterruptor8
CMP M[DefMeteorosCont],R0
BR.Z Interruptor81
MOV M[DefMeteorosCont],R0
MOV R2,VarDefinicoes9
MOV R1,111000101111b
CALL EscString ; Escreve alteracao no ecra
BR FimInterruptor8
Interruptor81: INC M[DefMeteorosCont]
MOV R2,VarDefinicoes8

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 5/18

```

MOV R1,111000101111b
CALL EscString ; Escreve alteracao no ecrã
FimInterruptor8: POP R1
RTI

;INT9
;
; Entradas: ---
; Sidas: ---
; Efeitos: Altera definicoes
Interruptor9: PUSH R1
MOV R1,3
CMP M[EstadoJogo],R1 ; Caso o estado do jogo seja def
inicoes, altera as definicoes associadas ao I9
BR.NZ FimInterruptor9
CMP M[DefMeteorosVel],R0
BR.Z Interruptor91
MOV M[DefMeteorosVel],R0
MOV R2,VarDefinicoes5
MOV R1,1000000101111b
CALL EscString ; Escreve alteracao no ecrã
BR FimInterruptor9
Interruptor91: INC M[DefMeteorosVel]
MOV R2,VarDefinicoes6
MOV R1,1000000101111b
CALL EscString ; Escreve alteracao no ecrã
FimInterruptor9: POP R1
RTI

;Temporizador
;
; Entradas: ---
; Sidas: ---
; Efeitos: Altera ContadorMissil
Temporizador: INC M[ContadorMissil] ;Contador para novo missil
INC M[MissilContador] ;Contador para mover o missil
INC M[ContGeraMeteoro] ;Contador para gerar um meteoro
INC M[ContMoveMeteoro] ;Contador para fazer mover os me
teoros

PUSH R1
MOV R1,TimeLong
MOV M[TimerValue],R1
MOV R1,EnableTimer
MOV M[TimerControl],R1
POP R1
RTI

//////////
; FUNCOES AUX ;
//////////

; MoveEsq: Move o canhao para a esquerda
; Entradas: ---
; Sidas: ---
; Efeitos: Modifica R1,R6,M[PosicaoRobot]
MoveEsq: MOV R1,0001011000000001b
CMP M[PosicaoRobot],R1
BR.Z FimMoveEsq
DEC M[PosicaoRobot]
CALL DrawRobot
MOV R6,M[PosicaoRobot]
INC R6
INC R6
MOV R1,' '

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 6/18

```

I2OP R1,R6
MOV M[FlagMoveCanhao],R2
FimMoveEsq: RET

; MoveDir: Move o canhao para a direita
; Entradas: ---
; Sidas: ---
; Efeitos: Modifica R1,R6,M[PosicaoRobot]
MoveDir: MOV R1,0001011001001110b
CMP M[PosicaoRobot],R1
BR.Z FimMoveDir
INC M[PosicaoRobot]
CALL DrawRobot
MOV R6,M[PosicaoRobot]
DEC R6
DEC R6
MOV R1,' '
I2OP R1,R6
MOV M[FlagMoveCanhao],R6
FimMoveDir: RET

//////////
; IMPRESSAO NO ECRã ;
//////////

; EscString: Escreve uma string para o ecrã
; Entradas: ---
; Sidas: ---
; Efeitos: altera registo R1, R2,R3,R4
EscString: MOV R3,FIM_TEXTO
CicloString: MOV R6,R1
I2OP M[R2],R6
INC R1
INC R2
CMP M[R2],R3
BR.NZ CicloString
RET

; DrawSet: Desenha o cenario
; Entradas: ---
; Sidas: ---
; Efeitos: Desenha o chao no ecrã
DrawSet: MOV R6,23
ROL R6,8
PUSH R5
PUSH R3
MOV R5,R0
MOV R3,SimboloChao
I2OP R3,R6
INC R6
INC R5
CMP R5,80
BR.NZ CicloSet
POP R3
POP R5
RET

; DrawRobot: Desenha o robot no cenario
; Entradas: ---
; Sidas: ---
; Efeitos: Desenha o robot no ecrã
DrawRobot: MOV R6,M[PosicaoRobot]

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 7/18

```

DEC     R6
MOV     R4,'<'
I2OP    R4,R6
INC     R6
MOV     R4,'^'
I2OP    R4,R6
INC     R6
MOV     R4,'>'
I2OP    R4,R6
RET

; DrawMeteoro:  Desenha o meteoro no ecra
;
; Entradas:    R2
; Saidas:      ---
; Efeitos:     Desenha meteoro no ecra
DrawMeteoro:  PUSH    R1
              MOV     R1,SimboloMeteoro
              I2OP    R1,M[R2]
              POP     R1
              RET

; LimpaEcra:    Coloca o simbolo espaco ' ' em todas as posicoes do ecra
;
; Entradas:    ---
; Saidas:      ---
; Efeitos:     Modifica o ecra todo
LimpaEcra:    MOV     R1,R0
CicloLimpaEcra:  MOV     R2,EspacoLimpaEcra
              CALL    EscString
              AND     R1,FF00h
              ADD     R1,0100h
              CMP     R1,1800h
              BR.NZ   CicloLimpaEcra
              RET

;;;;;;;;;;;;;
; IMPRESSAO NO 7SEG ;
;;;;;;;;;;;;;

; Print7Seg:    Escreve a pontuacao no display de 7 segmentos
;
; Entradas:    ---
; Saidas:      ---
; Efeitos:     ---
Print7Seg:    MOV     R7,M[LastScore]
              MOV     R3,Segmento71
ContPrint7Seg:  CMP     R7,R0
              BR.Z    ZeroPontos7Seg
              MOV     R6,0Ah
              DIV     R7,R6
              MOV     R4,R6
              ADD     R4,'0'
              MOV     M[R3],R4
              INC     R3
              BR      ContPrint7Seg
ZeroPontos7Seg:  RET

;;;;;;;;;;;;;
; IMPRESSAO NO LCD ;
;;;;;;;;;;;;;

; PrintLCD:     Escreve no LCD

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 8/18

```

; Entradas:    R1: Inicio Texto, R2, Posicao
; Saidas:      ---
; Efeitos:     ---
PrintLCD:    MOV     R3,FIM_TEXTO
CicloLCD:    CMP     M[R1],R3
              BR.Z    FimLCD
              MOV     R4,M[R1]
              MOV     M[LCD_Control],R2
              MOV     M[LCD_WRITE],R4
              INC     R1
              INC     R2
              BR      CicloLCD
FimLCD:      RET

; UpdateLCD1:   Atualiza pontuacao no LCD
;
; Entradas:    ---
; Saidas:      ---
; Efeitos:     ---
UpdateLCD1:  MOV     R7,M[Score]
              MOV     R3,800bh
ContAtualizaS:  CMP     R7,R0
              BR.Z    ZeroPontosAS
              MOV     R6,0Ah
              DIV     R7,R6
              MOV     R4,R6
              ADD     R4,'0'
              MOV     M[LCD_Control],R3
              MOV     M[LCD_WRITE],R4
              DEC     R3
              BR      ContAtualizaS
ZeroPontosAS:  RET

; UpdateLCD2:   Atualiza pontuacao maxima no LCD
;
; Entradas:    ---
; Saidas:      ---
; Efeitos:     ---
UpdateLCD2:  MOV     R7,M[MaxScore]
              MOV     R3,801bh
ContAtualizaS2:  CMP     R7,R0
              BR.Z    ZeroPontosAS2
              MOV     R6,0Ah
              DIV     R7,R6
              MOV     R4,R6
              ADD     R4,'0'
              MOV     M[LCD_Control],R3
              MOV     M[LCD_WRITE],R4
              DEC     R3
              BR      ContAtualizaS2
ZeroPontosAS2:  RET

;;;;;;;;;;;;;
; MISSIL ;
;;;;;;;;;;;;;

; TesteMissil   Verifica se e possivel enviar missil, caso positivo, flag a 1
;
; Entradas:    ---
; Saidas:      ---
; Efeitos:     Modifica R1,R2,M[FlagMissilReady],LED's
TesteMissil:  MOV     R1,44
              CMP     M[ContadorMissil],R1
              BR.N    TesteMissil2 ; Se o contador da flag do missi
1 for menos que 44, entao atualiza LEDs
              MOV     R1,1

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 9/18

MOV     M[FlagMissilReady],R1    ; Se o contador da flag do missi
1 for maior que 44, entao FlagMissilReady=1
MOV     R1,ffffh                 ; Atualiza valor a colocar nos L
EDs
BR       TesteMissil1
TesteMissil2: MOV     R2,R0          ; Se o contador <44, entao ve qu
al o valor a colocar nos LEDs
CicloMissil: MOV     R1,0000000000000000b
CMP      M[ContadorMissil],R2
BR.N     TesteMissil1
ROL      R1,1
INC      R1
ADD      R2,3
BR       CicloMissil
TesteMissil1: MOV     M[FFF8h],R1    ; Atualiza LEDs
RET

; LancaMissil: Caso a flag LancaMissil esteja a 1, lanca o missil
; Entradas: ---
; Saidas: ---
; Efeitos: Altera R1,R2,R4,M[FlagLancaMissil],M[ContadorMis
sil],M[MissilPosicao],M[MissilContador], posicao do missil no ecrã
LancaMissil: CMP      M[FlagLancaMissil],R0 ; Verifica se e possivel jogador
lancou o missil
BR.Z     FimLancaMissil ; Se nao lancou, fim
MOV      M[FlagLancaMissil],R0 ; Se lancou, limpa Flag
MOV      R1,1
MOV      M[ContadorMissil],R0 ; Repoe Contador missil
MOV      M[LEDControl],R0 ; Limpa LEDs
MOV      R2,M[PosicaoRobot]
ROL      R2,8
DEC      R2
ROL      R2,8
MOV      M[MissilPosicao],R2 ; Coloca a posicao do missil uma
linha acima da posicao do robot na mesma coluna
MOV      M[MissilContador],R0 ; Contador a zero
MOV      R1,M[SimboloMissil]
MOV      R6,R2
MOV      R7,SimboloMissil
I2OP     R7,R6 ; Desenha Missil
FimLancaMissil: RET

; MoveMissil: Move o missil
; Entradas: ---
; Saidas: ---
; Efeitos: Altera posicao do missil, e altera contador
MoveMissil: PUSH     R1
CMP      M[MissilPosicao],R0 ; Se nao existe missil, fim da r
otina
BR.Z     MoveMissil1
MOV      R1,M[MissilPosicao] ; Se existe, decrementa-se uma l
inha a posicao
SUB      R1,0000000010000000b
BR.NN    MoveMissil2
MOV      R2,M[MissilPosicao]
MOV      R4,' '
I2OP     R4,R2
MOV      M[MissilPosicao],R0
BR       FimMoveMissil
MoveMissil1: BR       FimMoveMissil
MoveMissil2: MOV      R1,2
CMP      M[MissilContador],R1
BR.N     FimMoveMissil

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 10/18

PUSH     R2
MOV      R2,M[MissilPosicao]
MOV      R4,' '
I2OP     R4,R2 ; Desenha.se um espaco na posica
o ocupada anteriormente
ROL      R2,8
DEC      R2
ROL      R2,8
MOV      M[MissilPosicao],R2 ; Muda posicao na memoria
MOV      M[MissilContador],R0 ; Limpa contador
MOV      R4,'|'
I2OP     R4,R2 ; Desenha nova posicao do missil
POP      R2
FimMoveMissil: POP     R1
RET

//////////
; METEORO ;
//////////

; GeraMeteoro: Gera um meteoro
; Entradas: ---
; Saidas: ---
; Efeitos: ---
GeraMeteoro: CMP      M[DefMeteorosQt],R0
BR.NZ    GeraMeteoro2
CMP      M[QtMeteoros],R0
JMP.NZ   FimGeraMeteoro
BR       GeraMeteoro3
GeraMeteoro2: MOV      R1,M[TimeGeraMeteoro]
CMP      M[DefMeteorosCont],R0
CALL.NZ  FuncaoAux1
CMP      M[ContGeraMeteoro],R1
JMP.N    FimGeraMeteoro
GeraMeteoro3: MOV      M[ContGeraMeteoro],R0
MOV      R6,5
CMP      M[DefMeteorosVel],R0
CALL.Z   ContAleatoria
MOV      M[TimeGeraMeteoro],R6
INC      M[QtMeteoros]
CALL     PosAleatoria
MOV      R1,R6
MOV      R2,R0
CALL     VelAleatoria
MOV      R3,R6
MOV      R4,MeteoroPos
CicloLM: CMP      M[R4],R0
BR.Z     SaltoLM
ADD      R4,3
BR       CicloLM
SaltoLM: MOV      M[R4],R1
INC      R4
MOV      M[R4],R2
INC      R4
MOV      M[R4],R3
SUB      R4,2
MOV      R2,R4
CALL     DrawMeteoro
FimGeraMeteoro: RET

FuncaoAux1: MOV      R1,50
RET

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 11/18

; MoveMeteoros: Ciclo que verifica se meteoro precisa de ser movido
; Entradas: ---
; Saidas: ---
; Efeitos: ---

MoveMeteoros: MOV R1,M[QtMeteoros]
MOV R2,MeteoroPos
Ciclo: CMP R1,R0 ; Repete o ciclo enquanto nao se
verificarem todos os meteoros
BR.Z Fim
CMP M[R2],R0 ; Caso a posicao seja zero, avan
ca 3 posicoes na memoria
BR.Z IncR21 ; E repete o ciclo
MOV R5,M[R2+2]
CMP R5,M[R2+1] ; Se contador=velocidade chama a
rotina para move um meteoro especifico
CALL.Z MoveMeteoro
DEC R1
IncR21: ADD R2,3
BR Ciclo
Fim: RET

; MoveMeteoro: Move o meteoro
; Entradas: R2
; Saidas: ---
; Efeitos: ---
MoveMeteoro: PUSH R1
PUSH R7
PUSH R4
MOV R1,M[R2] ; Apaga o meteoro existente no e
cra
MOV R4,SimboloEspaco ;
I2OP R4,R1
MOV R1,100h
ADD M[R2],R1 ; Altera a posicao do meteoro na
memoria
MOV M[R2+1],R0 ; Reinica contador do meteoro
MOV R7,M[R2]
SHR R7,8
CMP R7,10110b ; Verifica se Meteoro chegou a T
erra
BR.NZ ContinuaMove ; Se nao chegou, desenha o meteo
ro
LimpaMeteoro: MOV M[R2],R0 ; Se chegou, apaga o meteoro da
memoria
MOV M[R2+1],R0
MOV M[R2+2],R0
DEC M[QtMeteoros]
INC M[UserPoints]
MOV R4,800Fh
MOV M[LCD_Control],R4
MOV R4,M[UserPoints]
ADD R4,'0'
MOV M[LCD_WRITE],R4
BR SaltoMM
ContinuaMove: CALL DrawMeteoro ; Se nao chegou a Terra, desenha
o Meteoro
SaltoMM: POP R4
POP R7
POP R1
RET

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 12/18

; IncContMet: Incrementa os contadores dos meteoros
; Entradas: ---
; Saidas: ---
; Efeitos: Altera os contadores dos meteoros (memoria)
IncContMet: CMP M[ContMoveMeteoro],R0
BR.Z FimIncMet
MOV M[ContMoveMeteoro],R0
MOV R1,MeteoroPos
MOV R2,M[QtMeteoros]
CicloIncMet: CMP R2,R0
BR.Z FimIncMet
CMP M[R1],R0
BR.Z IncMet2
INC M[R1+1]
IncMet: ADD R1,3
DEC R2
BR CicloIncMet
IncMet2: ADD R1,3
BR CicloIncMet
FimIncMet: RET

;;;;;;;;;;;;;
; DESTRUIR ;
;;;;;;;;;;;;;

; VerificaM: Verifica se missil destruir meteoro
; Entradas: ---
; Saidas: ---
; Efeitos: ---
VerificaM: PUSH R1
PUSH R2
PUSH R3
PUSH R6
CMP M[MissilPosicao],R0 ; Se nao existe missil, ignora r
otina
JMP.Z FimVerificaM
MOV R1,M[MissilPosicao]
MOV R2,MeteoroPos
MOV R3,M[QtMeteoros]
CicloVerificaM: CMP R3,R0 ; Enquanto nao verifica todos os
meteoros existentes repete o ciclo
JMP.Z FimVerificaM ; Se ja verificou todos, termina
rotina
CMP M[R2],R0 ; Se posicao selecionada Ã zero
JMP.Z IncR2_1
CMP M[R2],R1 ; Se o conteudo da memoria for d
iferente da posicao do missil
JMP.NZ IncR2
MOV M[MissilPosicao],R0 ; Se for igual a posicao do miss
il, limpa a posicao do missil
MOV M[MissilContador],R0 ; Limpa o contador do missil
PUSH R2
MOV R2,SimboloEspaco
I2OP R2,R1 ; Desenha um espaco na posicao d
o missil
POP R2
DEC M[QtMeteoros] ; Decrementa QtMeteoros
MOV M[R2],R0 ; Limpa memoria associada ao met
ero (Posicao)
MOV M[R2+1],R0 ; (contador)
MOV M[R2+2],R0 ; (velocidade)
CALL ScoreAleatorio ; Define uma pontuacao para o me

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 13/18
teoro destruido
MOV M[LastScore],R6 ;
ADD M[Score],R6 ;
CALL Print7Seg ; Escreve essa pontuacao no disp
lay 7seg
CALL UpdateLCD1 ; Atualiza pontuacao atual
JMP FimVerificaM ; Fim da rotina
IncR2_1: ADD R2,3 ; avanca 3 posicoes na memoria e
verifica novamente
JMP CicloVerificaM
IncR2: ADD R2,3 ; avanca na memoria
DEC R3 ; Decrementa contador que conta
os meteoros verificados
JMP CicloVerificaM
FimVerificaM: POP R6
POP R3
POP R2
POP R1
RET

//////////
; RANDOM ;
//////////

; ScoreAleatorio: Gera um valor de pontuacao aleatorio
; Entradas: ---
; Sidas: ---
; Efeitos: Valor em R6
ScoreAleatorio: PUSH R2
CicloScore: CALL Aleatorio
MOV R2,20
DIV R6,R2 ; Divide-se por 20
MOV R6,R2 ; Valor considerado e o resto da divisao
(sempr inferior a R2)
CMP R6,R0 ; Se for inferior a 0 repete-se a rotina
BR.NP CicloScore
ADD R6,10 ; Soma-se 10 porque valores gerados esta
o entre 0 e 20
POP R2
RET

; VelAleatoria: Gera um valor de velocidade aleatorio
; Entradas: ---
; Sidas: ---
; Efeitos: Valor em R6
VelAleatoria: PUSH R2
CicloVel: CALL Aleatorio
MOV R2,4
DIV R6,R2 ; Divide-se por 4
MOV R6,R2 ; Valor considerado e o resto da divisao
(sempr inferior a R2)
CMP R6,R0 ; Se for inferior a zero, repete-se a ro
tina
BR.NP CicloVel
POP R2
RET

; ContAleatoria: Gera um valor de tempo aleatorio
; Entradas: ---
; Sidas: ---
; Efeitos: Valor em R6
ContAleatoria: PUSH R2

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 14/18
CicloAleat: CALL Aleatorio
MOV R2,30
DIV R6,R2 ; Divide-se por 30
MOV R6,R2 ; Valor considerado e o resto da divisao
(sempr inferior a R2)
CMP R6,R0 ; Caso seja inferior a zero, repete-se a
rotina
BR.NN CicloAleat
ADD R6,10 ; Soma-se 10, porque valores gerados est
ao entre 0 e 30
POP R2
RET

; PosAleatoria: Gera uma posicao aleatoria
; Entradas: ---
; Sidas: ---
; Efeitos: Valor em R6
PosAleatoria: PUSH R2
CicloPosAleat: CALL Aleatorio
MOV R2,79
DIV R6,R2 ; Divide-se por 79 (posicoes na janela)
MOV R6,R2 ; Valor considerado e o resto da divisao
(sempr inferior a R2)
CMP R6,R0 ; Caso o resto nao seja maior que zero,
repete-se o algoritmo
BR.NP CicloPosAleat
POP R2
RET

; Aleatorio: Gera um valor aleatorio (algoritmo descrito no enunciado)
; Entradas: ---
; Sidas: ---
; Efeitos: Valor em R6
Aleatorio: MOV R6,M[LASTRANDOM]
AND R6,1
CMP R7,R0
BR.NZ SaltoAleatorio
MOV R6,M[LASTRANDOM]
ROR R6,1
MOV M[LASTRANDOM],R6
RET
SaltoAleatorio: MOV R6,M[LASTRANDOM]
XOR R6,M[RANDOM]
ROR R6,1
MOV M[LASTRANDOM],R6
RET

//////////
//////////
////// JOGO ////
//////////
//////////

Inicio: ENI ; Ativa interrupcoes
MOV R5,FFFFh ;
MOV M[IO_Control],R5 ; Ativa o selector de posicao no
ecra
CALL LimpaEcra ;
MOV R2,VarTexto1 ;
MOV R1,0000110000011001b ;
CALL EscString ; Escreve frase1 de introducao n
o ecra

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 15/18

o ecra
MOV R2,VarTexto2
MOV R1,0000111000010110b
CALL EscString ; Escreve frase2 de introducao n

o ecra
MOV R2,VarTexto3
MOV R1,0001000000010001b
CALL EscString ; Escreve frase2 de introducao n

o ecra
MOV R2,VarTexto4
MOV R1,0001001000010001b
CALL EscString ; Escreve frase2 de introducao n

EcraInicial: INC M[LASTRANDOM] ; Incrementa o LastRandom para g
erar valores complemente diferentes cada vez que corremos o programa
CMP M[EstadoJogo],R0 ; Se nao foi primido nenhum bota
o volta a verificar
BR.Z EcraInicial
MOV R1, 3
CMP M[EstadoJogo],R1 ; Se jogador primiu I3, mostra a
s definicoes
JMP.Z VerDefinicoes2
MOV R1,4
CMP M[EstadoJogo],R1 ; Se jogador primiu I2, mostra a
s instrucoes
JMP.Z Instrucoes
JMP ComecaJogo ; Volta ao ecra inicial

;;;;;;;;;;
; MOSTRA INSTRUCOES ;
;;;;;;;;;;

Instrucoes: CALL LimpaEcra ; Limpa o ecra
MOV R2,VarInstrucoes
MOV R1,0000100000001010b
CALL EscString ; Escreve string 1 no ec

ra
MOV R2,VarInstrucoes1
MOV R1,00001010000001010b
CALL EscString ; Escreve string 2 no ec

ra
MOV R2,VarInstrucoes2
MOV R1,00001100000001010b
CALL EscString ; Escreve string 3 no ec

ra
MOV R2,VarInstrucoes3
MOV R1,00001110000001010b
CALL EscString ; Escreve string 4 no ec

ra
MOV R2,VarInstrucoes4
MOV R1,00010000000001010b
CALL EscString ; Escreve string 5 no ec

ra
MOV R2,VarInstrucoes5
MOV R1,00010010000001010b
CALL EscString ; Escreve string 6 no ec

ra
CicloInstrucoes: CMP M[EstadoJogo],R0 ; Verifica se jogador pr
imiu I1 (Voltar)?
JMP.Z Inicio ; Se sim, volta ao ecra
inicial
BR CicloInstrucoes ; Se nao, volta a verifi
car

```

```

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 16/18

;;;;;;;;;;
; MOSTRA DEFINICOES ;
;;;;;;;;;;

VerDefinicoes2: CALL LimpaEcra
MOV R2,VarDefinicoes
MOV R1,0A0Ah
CALL EscString
MOV R2,VarDefinicoes1
MOV R1,0C0Ah
CALL EscString
MOV R2,VarDefinicoes2
MOV R1,100Ah
CALL EscString
MOV R2,VarDefinicoes7
MOV R1,0E0Ah
CALL EscString
CMP M[DefMeteorosQt],R0
BR.Z Escreve1
BR.NZ Escreve2
EscreveVel: CMP M[DefMeteorosVel],R0
BR.Z Escreve3
BR.NZ Escreve4
BR EscreveVel2
EscreveVel2: CMP M[DefMeteorosCont],R0
JMP.Z Escreve5
JMP.NZ Escreve6
JMP CicloDefinicoes
Escreve1: MOV R2,VarDefinicoes4
MOV R1,0C2Fh
CALL EscString
BR EscreveVel
Escreve2: MOV R2,VarDefinicoes3
MOV R1,0C2Fh
CALL EscString
BR EscreveVel
Escreve3: MOV R2,VarDefinicoes5
MOV R1,102Fh
CALL EscString
JMP EscreveVel2
Escreve4: MOV R2,VarDefinicoes6
MOV R1,102Fh
CALL EscString
JMP EscreveVel2
Escreve6: MOV R2,VarDefinicoes8
MOV R1,0E2Fh
CALL EscString
JMP CicloDefinicoes
Escreve5: MOV R2,VarDefinicoes9
MOV R1,0E2Fh
CALL EscString
BR CicloDefinicoes
CicloDefinicoes: CMP M[EstadoJogo],R0
JMP.Z Inicio
BR CicloDefinicoes

;;;;;;;;;;
; COMECA O JOGO ;
;;;;;;;;;;

ComecaJogo: CALL LimpaEcra ; Limpa o ecra

```


mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 17/18			
MOV	M[UserPoints],R0		; Repoe o numero de mete
oros que atingiram a terra			
MOV	M[Score],R0		; Repoe pontuacao
MOV	M[LastScore],R0		; Repoe pontuacao meteor
o			
MOV	M[FlagMoveCanhao],R0		; Repoe flag de move met
eoro			
MOV	R4,800Fh		
MOV	M[LCD_Control],R4		
MOV	R4,M[UserPoints]		
ADD	R4,'0'		; Atualiza o numero de m
eteoros que atingiram a terra no LCD			
MOV	M[LCD_WRITE],R4		
MOV	R1,1		
MOV	M[FlagMissilReady],R1		; Ativa flag MissilReady
MOV	M[FlagLancaMissil],R0		; Desativa flag Lancar M
issil			
MOV	R1,44		
MOV	M[ContadorMissil],R1		; Contador MissilReady a
44 (ativo)			
CALL	Print7Seg		; Desenha o display de 7
segmentos			
CALL	DrawSet		; Desenha cenario
CALL	DrawRobot		; Desenha robot
MOV	R1,VarPontos		
MOV	R2,1000000000000000b		
CALL	PrintLCD		; Atualiza LCD (pontos)
MOV	R1,VarMaximo		
MOV	R2,1000000000010000b		
CALL	PrintLCD		; Atualiza LCD (Highscor
e)			
CALL	UpdateLCD2		; Atualiza LCD (Numero d
e meteoros que atingiram a terra)			
MOV	R1,TimeLong		
MOV	M[TimerValue],R1		; Programa temporizador
MOV	R1,EnableTimer		
MOV	M[TimerControl],R1		; Ativa o temporizador
CicloJogo:	CMP	M[FlagMoveCanhao],R0	; Vefifica FlagMoveCanha
o			
CALL.Z	MoveEsq		; Se for 0 move o canhao
para a esquerda			
MOV	R1,1		
CMP	M[FlagMoveCanhao],R1		
CALL.Z	MoveDir		; Se for 1 move o canhao
para a direita			
CALL	TesteMissil		; Chama TesteMissil (que
acende os LEDs e ativa FlagMissilReady)			
CALL	LancaMissil		; Chama LancaMissil (que
lanca o missil caso I2 primido e FlagMissilReady = 1)			
CALL	MoveMissil		; Chama MoveMissil (Move
o missil caso tenha sido disparado)			
CALL	VerificaM		; Verifica se Missil ati
ngiu meteoros			
CALL	IncContMet		; Incrementa os contador
es dos meteoros individualmente			
CALL	MoveMeteoros		; Move os meteoros
CALL	VerificaM		; Verifica se Missil ati
ngiu meteoros			
CALL	GeraMeteoro		; Gera um novo meteoro s
e contador=velocidade			
MOV	R1,3		
CMP	M[UserPoints],R1		; Verifica se o jogador

mai 17, 13 14:35 D:\Users\Joao\Desktop\p3print_win\GnuWin32\bin\fase1.as Page 18/18			
perdeu?			
JMP.NN	FimJogo		; Se sim, Fim do Jogo
JMP	CicloJogo		; Se nao, volta ao Ciclo
Jogo			
;;;;;;;;;;			
; FIM DO JOGO ;			
;;;;;;;;;;			
FimJogo:	MOV	R1,M[Score]	
CMP	M[MaxScore],R1		; Compara pontuacao com
o Highscore			
BR.P	FimJogo2		; Se for inferior, nao a
tualiza			
MOV	M[MaxScore],R1		; Senao, atualiza
CALL	UpdateLCD2		; Escreve Highscore no L
CD			
FimJogo2:	MOV	R1,M[QtMeteoros]	; Limpa posicoes de Memo
ria dos meteoros			
CicloFimJogo3:	CMP	R1,R0	; Se ja tiverem sido lim
pos todos os meteoros, avanca			
BR.Z	FimJogo3		; Avanca
MOV	R2,MeteoroPos		; Senao verifica posicao
seguite			
CicloFimJogo2:	CMP	M[R2],R0	; Caso seja zero, avanca
na memoria			
BR.Z	FimJogo4		; Ignora verificacao e a
vanca na memoria			
MOV	M[R2],R0		; Senao, limpa a memoria
DEC	R1		; Decrementa o numero de
meteoros			
BR	CicloFimJogo3		; Volta a verificar
ADD	R2,3		; Incrementa a posicao d
FimJogo4:			
e memoria a verificar em 3 posicoes			
BR	CicloFimJogo2		; Volta a verificar
FimJogo3:	MOV	M[UserPoints],R0	; Limpa numero de meteor
os que atingiram a terra			
MOV	M[QtMeteoros],R0		; Limpa o numero de mete
oros existentes no ecra			
MOV	R1,5672		
MOV	M[PosicaoRobot],R1		; Repoe posicao do canha
o			
MOV	R1,2		
MOV	M[EstadoJogo],R1		; Altera o estado do jog
o			
CALL	LimpaEcra		; Limpa o ecra
MOV	R2,VarFimJogo1		
MOV	R1,B22h		
CALL	EscString		; Escreve frase2 de intr
oducao no ecra			
MOV	R2,VarFimJogo2		
MOV	R1,D16h		
CALL	EscString		; Escreve frase2 de intr
oducao no ecra			
CicloFimJogo:	MOV	R1,1	
CMP	M[EstadoJogo],R1		; Verifica se o jogador
primiu I1?			
BR.NZ	CicloFimJogo		; Se nao primiu, volta a
verificar			
JMP	ComecaJogo		; Se primiu, inicia o jo
go			