Este exame tem 7 questões, num total de 200 pontos. Responda em folhas separadas a cada um dos seguintes conjuntos de problemas: (1 e 2), (3 e 4), (5 e 6). O problema 7 deve ser respondido na folha de enunciado.

1. Considerar o seguinte programa escrito em linguagem *assembly* IA-32.

```
1  include   mpcp.inc
2            .data
3  valores   SDWORD 1, 2, 3, 5, 7, 11, 13, 17, 19 ;sequência ordenada
4  fmt       BYTE   "Resultado=%d Contador=%d", 13, 10, 0
5  contador  DWORD  ?
6
7  procurar  PROTO seq:PTR SDWORD, val:DWORD, inf:DWORD, sup:DWORD
8
9            .code
10 main PROC C
11         mov     contador, 0
12         invoke  procurar, offset valores, 13, 0, lengthof valores
13         invoke  printf, offset fmt, eax, contador
14         invoke  ExitProcess, 0
15 main ENDP
16
17 procurar PROC USES edi ebx seq:PTR SDWORD, val:DWORD,
18                           inf:DWORD, sup:DWORD
19          inc   contador    ; para contabilizar o número de invocações
20          mov   edi, seq
21          mov   ecx, val
22          mov   edx, inf
23         .IF  sup <= edx
24              cmp  ecx, [edi+4*edx]
25              jne  nao_existe
26              mov  eax, edx
27              jmp  final
28         .ELSE
29              add  edx, sup
30              shr  edx, 1
31              mov  ebx, [edi+4*edx]
32              cmp  ebx, ecx
33              jl   acima
34              jg   abaixo
35              mov  eax, edx
36              jmp  final
37         .ENDIF
38 abaixo:  invoke  procurar, seq, val, inf, edx
39          jmp     final
40 acima:   inc     edx
41          invoke  procurar, seq, val, edx, sup
42          jmp     final
43 nao_existe:
44          mov     eax, -1
45 final:   ret
46 procurar ENDP
47 END
```

[20] (a) Explicar o funcionamento da sub-rotina `procurar` e indicar o significado de cada parâmetro. Indicar e justificar o que é apresentado no monitor quando o programa é executado.

Resposta: A sub-rotina `procurar` é recursiva. Em todas as invocações recursivas, os argumentos `seq` e `val` mantêm os seus valores. Para cada invocação, a variável global `contador` é incrementada: no fim conterá o número de vezes que a sub-rotina `procurar` foi invocada (já que o valor inicial da variável é 0).

Quando existe uma invocação recursiva (linhas 38 ou 41, mas não ambas na mesma invocação), o resultado da invocação recursiva é também o resultado da invocação corrente (existe apenas um salto para o fim da rotina, sem alterar o valor de `EAX`).

Caso geral: Enquanto `inf`<`sup`,o valor de `edx` é colocado a `inf+sup)/2` (linha 29 e seguinte). Se o valor de índice `edx` da sequência for menor que `val` é executada a invocação referenciada por `abaixo`, que repete o procedimento para o intervalo inferior (entre os índices `inf` e `edx`). Se o valor de índice `edx` da sequência for maior que `val` é executada a invocação referenciada por `acima`, que repete o procedimento para o intervalo superior (entre os índices `edx+1` e `sup`). Caso o valor de índice `edx` da sequência seja igual `val`, a sub-rotina termina com o resultado igual a `edx`. Notar que o limite superior não faz parte do intervalo em análise `[inf,sup[`.

Caso base: A recursão termina se forem executadas as linhas 27 ou 36 . Isso acontece quando a relação `inf` < `sup` deixa de se verificar (linha 23). Nesse caso, se o valor da sequência com índice `inf` for diferente de `val` a sub-rotina retorna o valor -1; senão, retorna o valor `inf`, que é o índice de `val` em `seq`.

Em conclusão, a sub-rotina `procurar` implementa uma versão recursiva da pesquisa binária, retornando o índice de `seq` onde está o valor `val` se o encontrar, ou -1 caso não exista. Em cada invocação, a pesquisa é feita no intervalo de índices `[inf,sup[`.

Para o programa apresentado, o valor 13 está na posição 6 da sequência. Os índices usados na pesquisa são 4 (=(0+9)/2), 7 (=(5+9)/2) e 6 (=(5+7)/2), sendo que o valor é encontrado nesta última posição. Logo, ocorrem três invocações da sub-rotina. No monitor é, portanto, apresentada uma linha: `Resultado=6 Contador=3`

[10] (b) Justificar a seguinte afirmação:

Para a sequência `seq` *apresentada, o valor de* `contador` *é sempre menor ou igual a 4.*

Resposta: Quando o valor procurado está na sequência, o comprimento da parte da sequência a pesquisar é sucessivamente reduzido a metade. Logo, o espaço de pesquisa é subdividido, no máximo, $\lceil \log_2(n) \rceil$ vezes, em que n é o número de elementos da sequência. Como $\lceil \log_2(9) \rceil = 4$, o contador de invocações não será superior a 4.

Notar que se o valor não existir, pode ser necessária ainda uma última invocação para determinar que o espaço de pesquisa está esgotado (o primeiro teste da sub-

rotina). Nesse caso, o número máximo de invocações é $\lceil \log_2(n) \rceil + 1$. Para $n = 9$, o valor da expressão é 5 e a afirmação não seria verdadeira.
(Observação: Ambas as respostas receberam cotação completa.)

[10] (c) Assumir que a pilha tem 64 bytes livres quando a sub-rotina `procurar` é invocada com os valores indicados no programa. O programa funcionaria corretamente? Justificar.

Resposta: Cada invocação necessita de uma moldura de 8×4 bytes (quatro DWORDs para os argumentos, uma DWORD para o endereço de retorno, uma DWORD para guardar o valor de `EBP` e duas DWORDs para preservar os registos). Com 64 bytes livres, a pilha não tem espaço para as 4 molduras que podem ser necessárias, pelo que o programa poderia não funcionar corretamente.

2. Uma sub-rotina sem variáveis locais tem o seguinte epílogo:

```
pop  edi
leave
ret  8
```

O estado da pilha imediatamente antes de executar o epílogo é o apresentado na tabela (valores apresentados em formato hexadecimal).

Endereço	Conteúdo
004AC05C	04210299
004AC058	53262105
004AC054	0001FC44
004AC050	0A11115F
004AC04C	09BF00F0
004AC048	004ACEE8
004AC044	80000000

[10] (a) Indique o significado que atribui ao conteúdo `80000000`.

Resposta:
Como a instrução `pop edi` recupera o valor de `EDI` guardado na pilha aquando da execução do prólogo, conclui-se que `80000000` era o valor de `EDI` antes da execução da sub-rotina.

[10] (b) Mostre qual o endereço do topo da pilha antes da invocação da sub-rotina.

Resposta:
No endereço `004AC048` está o valor de EBP, resultante da primeira instrução do prólogo. Em `004AC04C` encontra-se o endereço de retorno da sub-rotina. Nas posições `004AC050` e `004AC054` estão os dois argumentos passados à sub-rotina quando foi invocada (conclui-se serem dois porque `ret 8` indica que são libertados da pilha 8 bytes referentes aos argumentos). Desta forma, o topo da pilha antes da invocação da sub-rotina tem o endereço `004AC058`.

[20] 3. Um sistema composto por um CPU, que opera a 1 GHz e um disco duro que transfere dados em grupos de 4 palavras (8 bytes cada) a uma taxa de 8 MB/s, utiliza o método de comunicação com periféricos conhecido como interrupções. Assumindo que o *overhead* de

cada transferência, incluindo o atendimento da interrupção, é de 2000 ciclos de relógio e que o disco duro transfere dados durante 10 % do tempo, calcule a percentagem de tempo médio de CPU consumido nas transferências. [Considere kB = 10^3 B, MB = 10^6 B.]

Resposta: Dados transferidos por acesso:

$$4 \times 8\text{B} = 32\,\text{B}$$

Acessos por segundo:

$$\frac{8\,\text{MB/s}}{32\text{B}} = \frac{1 \times 10^6}{4 \times 8} = 0{,}25 \times 10^6$$

Uma vez que o disco só transfere dados em 10% do tempo o numero de acessos será de:

$$0{,}10 \times 0{,}25 \times 10^6 = 25 \times 10^3$$

Então o número de ciclos consumidos pela operação será de:

$$25 \times 10^3 \ \times 2 \times 10^3 = 5 \times 10^7$$

Tento em conta o número de ciclos consumidos por segundo, a percentagem média de tempo de CPU consumida pela técnica será de:

$$\frac{5 \times 10^7}{1 \times 10^9} = 5 \times 10^{-2} = 0{,}05 = 5\,\%$$

4. O computador de bordo de uma boia oceanográfica envia por rádio, a cada n minutos, a posição da boia e um conjunto de informações meteorológicas relevantes, num total de 2500 Bytes. O rádio tem uma potência de 180 W e envia informação a uma cadência de 10 kbit/s; o restante `hardware` tem uma potência de 1 W. A bateria tem capacidade para armazenar 1200 W h de energia. A boia é lançada ao mar com a bateria totalmente carregada.

[10] (a) Determine qual deve ser a periodicidade do envio de informação (valor de n) por forma a que a boia possa navegar durante 25 dias. Note que o número de envios por hora é dado por $60/n$.

Resposta:

Tempo de envio de 2500 Bytes: $\frac{2500 \times 8}{10 \cdot 10^3} = \frac{20 \cdot 10^3}{10 \cdot 10^3} = 2\text{s} = \frac{2}{3600}\text{h}$

Energia por envio: $180 \times \frac{2}{3600} = 0{,}1\,\text{W h}$; envios por hora: $\frac{60}{n}$

Energia por hora: $1 + 0{,}1 \times \frac{60}{n} = \left(1 + \frac{6}{n}\right)\text{W h}$

Em 25 dias: $25 \times 24 \times \left(1 + \frac{6}{n}\right) = 1200 \Leftrightarrow 1 + \frac{6}{n} = 2 \Leftrightarrow n = 6$

A periodicidade será pois um envio a cada 6 minutos, isto é, 10 envios por hora.

[10] (b) A cadência de envio da informação é agora de minuto a minuto e a boia foi equipada com painéis fotovoltaicos capazes de fornecer, em média, 160 W h de energia por dia. Determine quantos dias a boia se manterá em funcionamento.

Resposta:

Energia por envio: $180 \times \frac{2}{3600} = 0{,}1\,\text{W h}$; envios por hora: 60

Energia por hora: $1 + 0.1 \times 60 = 7\text{W h}$

Em 24h, gasta: $24 \times 7 = 168\,\text{W h}$; reposta: 160W h; saldo: 8W h gastos por dia

Duração da bateria: $\frac{1200}{8} = 150$ dias

A boia poderá navegar durante 150 dias.

[30] 5. O volume de um cone de altura h é dado pela expressão:

$$V = \frac{A_b \times h}{3} \qquad \text{com} \quad A_b = \pi \times r^2$$

em que A_b representa a área da base de raio r.

Implemente a sub-rotina `vcone` com o protótipo

```
vcone PROTO C r:REAL8, h:REAL8, vol:PTR REAL8
```

que calcula o volume de um cone, guardando o resultado na posição indicada pelo apontador `vol`. Para o cálculo da área da base pode assumir que existe uma variável global `PI` do tipo `REAL8` que contém o valor de π. Não devem ser usadas outras variáveis globais.

Resposta:

```
vcone PROC C r:REAL8, h:REAL8, vol:PTR REAL8
        movsd xmm0, r
        mulsd xmm0, xmm0
        mulsd xmm0, PI
        mulsd xmm0, h
  mov   eax, 3
        cvtsi2sd xmm1, eax
        divsd xmm0, xmm1
        mov   eax, vol
        movsd REAL8 PTR [eax], xmm0
        ret
vcone ENDP
```

6. Cada uma das seguintes questões tem apenas uma resposta certa. Indique as respostas corretas **na folha de resposta** (e não na folha do enunciado).

[5] (a) Um sistema RAID-5 com capacidade *útil* de 100 TB está organizado em 10 grupos de proteção. De entre as seguintes alternativas, selecione a capacidade de cada disco de forma a obter a melhor fiabilidade. Assumir que a fiabilidade de todos os discos é igual.

A. 5 TB B. 2 TB C. 500 GB D. 1 TB

[5] (b) Qual das alterações indicadas a seguir **não** aumenta a disponibilidade de um sistema?

A. Reduzir o MTTR. B. Reduzir o MTTF.

C. Usar um processador mais rápido.

D. Substituir fontes de alimentação normais por fontes redundantes.

[5] (c) Qual é o protótipo em linguagem *assembly* da sub-rotina que tem a seguinte declaração em C++:

```
extern "C" int func(char nome[], unsigned int tam, double valores[])
```

A. `func PROTO C nome:BYTE, tam:DWORD, valores:REAL8`

B. `func PROTO C nome:PTR BYTE, tam:DWORD, valores:PTR REAL8`

C. `func PROTO C nome:PTR BYTE, tam:WORD, valores:PTR REAL4`

D. `func PROC C uses EAX nome:BYTE, tam:DWORD, valores:PTR REAL8`

[5] (d) Assumir que `ECX=4`, `AX=0`, `EDI=0000A000h` e `DF=1`. Qual é a gama de endereços cujo conteúdo é colocada a zero pela instrução `rep stosw` ?

A. `[0000A000h; 0000A007h]` B. `[00009FF9h; 0000A000h]`

C. `[00009FFAh; 0000A001h]` D. `[00009FE0h; 0000A000h]`

[5] (e) Considerar a sub-rotina cujo cabeçalho é o seguinte:

```
funcX PROC uses ebx arg1:SWORD, arg2:SWORD
```

Qual das seguintes instruções é legal quando usada no código de `funcX`?

A. `mov ebx, arg2` B. `mov ebx, offset arg1`

C. `cmp arg1, arg2` **D.** `lea ebx, arg2`

[5] (f) Indique a informação verdadeira sobre variáveis locais.

A. Podem ser usadas em todo o código do mesmo módulo (ficheiro).

B. Fazem uso eficiente da memória porque o seu espaço em memória pode ser libertado quando a sub-rotina termina.

C. Tipicamente, são criadas no segmento de dados.

D. No mesmo ficheiro, todas elas devem ter nomes diferentes.

Nome (legível): _____

7. Uma sequência de valores crescentes do tipo `dword` e terminada por zero representa as idades dos indivíduos de uma população. Por exemplo $\{8, 10, 13, 14, 17, 47, 0\}$ representa as idades de uma população com 6 elementos. Recorde que a mediana é o valor que tem tantos elementos da população inferiores a ele como superiores a ele; ou seja, é o elemento central de uma sequência ordenada com número ímpar de elementos ou a média dos elementos centrais se o número de elementos for par. No exemplo apresentado a mediana é 13,5.

[20] (a) Escreva a rotina `mediana proto seq:ptr dword, res:ptr real8` que calcula a mediana de uma população. Nota: `seq` aponta o primeiro elemento da sequência e `res` aponta o resultado.

Resposta:

```
mediana proc uses esi seq: ptr dword, res: ptr real8
       mov esi, seq
       xor ecx, ecx      ;; contador de elementos

@@:    mov eax, [esi]
       cmp eax, 0
       jz  acaba         ;; chegou ao fim da sequencia
       add esi, 4
       inc ecx
       jmp @b

acaba: mov esi,seq
       rcr ecx,1        ;; testa se é par ou impar
       jnc par

impar: cvtsi2sd xmm0, dword ptr[esi+4*ecx]    ;; Elemento central
       jmp fim

par:   cvtsi2sd xmm0, dword ptr[esi+4*ecx]    ;; Elemento central 2
       cvtsi2sd xmm1, dword ptr[esi+4*ecx-4]  ;; Elemento central 1
       addsd xmm0, xmm1                       ;; Soma em xmm0
       mov eax, 2
       cvtsi2sd xmm1, eax
       divsd xmm0, xmm1                       ;; Média em xmm0

fim:   mov esi,res                            ;; Aponta resultado
       movsd real8 ptr [esi], xmm0            ;; Guarda resultado
       ret
mediana endp
```

Nota:

Também se poderia ter usado SPFP nos acessos aos elementos centrais mas no fim seria sempre necessário converter para DPFP pois o resultado final é do tipo `real8`.

[20] (b) Complete nos locais assinalados o programa principal que imprime a população e a sua mediana. No exemplo considerado o programa deve escrever:

```
Amostra: 10 10 13 14 17 47
Mediana: 13.5
```

Código a completar:

```
include mpcp.inc
.xmm

mediana proto seq: ptr dword, res:ptr real8

.data
pop1  dword 10, 10, 13, 14, 17, 47, 0
med   real8 0.0
msg1  byte "Amostra: ",0
msg2  byte "%d ",0
msg3  byte 10,13,"Mediana: %lf",0

.code
main: invoke printf, offset msg1
      mov esi, offset pop1
      mov ecx, lengthof pop1
      dec ecx

@@:   push ecx
      invoke printf, offset msg2, dword ptr[esi]
      add esi,4
      pop ecx
      loop @b

      invoke mediana, offset pop1, offset med
      invoke printf, offset msg3, med

      invoke _getch
      invoke ExitProcess, 0
end main
```