


Pergunta 1



Apresenta-se a seguir o código máquina, e o correspondente código de montagem, para 3 instruções de um programa que está em execução através do **gdb**.


```
...  
8048135: ff 00      incl (%eax)  
8048137: eb b3      jmp ???  
8048139: 8b 55 f4    mov 0xffffffff4(%ebp), %edx  
...
```


Se o programa estiver num ponto de paragem, e os valores dos registos forem `eip=0x8048137` e `esp=0xbffc00cb`, após executar um comando `stepi` o novo valor de **eip** é 0x **[v1EIP]**.

Resposta especificada para: v1EIP  80480ec

Respostas corretas para: v1EIP

Método de avaliação

 Correspondência de padrão

 Correspondência de padrão

Resposta correta

`\s*(?i)(0*80480ec)\s*`

`\s*(?i)(0x0*80480ec)\s*`

Sensibilidade a maiúsculas e minúsculas

Pergunta 2



Apresenta-se a seguir o código máquina, e o correspondente código de montagem, para 3 instruções de um programa que está em execução através do **gdb**.

```
...
804815d: 50          push %eax
804815e: e8 62 ff ff call 0x80480ce
8048163: 83 c4 10    add $0x10,%esp
...
```

Se o programa estiver num ponto de paragem, e os valores dos registos forem `eip=0x804815e`, `esp=0xbfff1d30`, e o conteúdo da memória for `M[0xbfff1d2c]=0`, `M[0xbfff1d30]=0`, `M[0xbfff1d34]=0`, após executar um comando *stepi* o valor de **eip** é 0x **[v2EIP]**, o **esp** é 0x **[v2ESP]**, os endereços de memória alterados vão de 0x **[memA]** a 0x **[memB]**, e o valor escrito nesses endereços é 0x **[memAval]**.

Resposta especificada para: v2EIP ✖ 80480ce

Resposta especificada para: v2ESP ✖ bfff1d3c

Resposta especificada para: memA ✔ bfff1d2c

Resposta especificada para: memB ✖ bfff1d3c

Resposta especificada para: memAval ✖ 00

Respostas corretas para: v2EIP		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0*804815e)\s*</code>	
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0x0*804815e)\s*</code>	
Respostas corretas para: v2ESP		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0*bfff1d2c)\s*</code>	
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0x0*bfff1d2c)\s*</code>	
Respostas corretas para: memA		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0*bfff1d2c)\s*</code>	
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0x0*bfff1d2c)\s*</code>	
Respostas corretas para: memB		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0*bfff1d2f)\s*</code>	
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0x0*bfff1d2f)\s*</code>	
Respostas corretas para: memAval		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0*8048163)\s*</code>	
✔ <i>Correspondência de padrão</i>	<code>\s*(?i)(0x0*8048163)\s*</code>	

Pergunta 3



Complete o fragmento de código em C com *valores, operações e variáveis*, tendo em atenção o código de montagem gerado pelo compilador.

```
int f( int *s, int v, int c){
    int resultado= [reti];
    int i;

    for ( i= [bi] ; i < [ei] ; i+= [ii] )
    {
        resultado [op]= s[i];
    }

    return resultado [opr] [vr];
}
```

```
f:
    pushl    %ebp
    movl     %esp, %ebp
    movl     12(%ebp), %ecx
    movl     16(%ebp), %edx
    cmpl     %ecx, %edx
    pushl    %ebx
    movl     $2, %eax
    movl     8(%ebp), %ebx
    jge      .L8
.L6:
    imull    (%ebx,%edx,4), %eax
    addl     $5, %edx
    cmpl     %ecx, %edx
    jl       .L6
.L8:
    addl     $2, %eax
    popl     %ebx
    leave
    ret
```

Resposta especificada para: reti ☒ 2

Resposta especificada para: bi ☒ c

Resposta especificada para: ei ☒ v

Resposta especificada para: ii ☒ 5

Resposta especificada para: op ☒ *

Resposta especificada para: opr ☒ +2

Resposta especificada para: vr ☒ .s[0]

Respostas corretas para: reti

Método de avaliação

☒ Correspondência de padrão

Resposta correta

\s*2\s*

Sensibilidade a maiúsculas e minúsculas

Respostas corretas para: bi		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*c\s*</code>	
Respostas corretas para: ei		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*v\s*</code>	
Respostas corretas para: ii		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*5\s*</code>	
Respostas corretas para: op		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*1*\s*</code>	
Respostas corretas para: opr		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*1+\s*</code>	
Respostas corretas para: vr		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	<code>\s*2\s*</code>	

Pergunta 4



O procedimento funcao (com algumas expressões omitidas) trabalha com o tipo de dados sistemascomputacao.

```
struct sistemascomputacao{
    int a[4];
    short b[2];
    struct {
        int x;
        int y;
    } s;
};
```

Qual o tamanho da estrutura sistemascomputacao (em bytes, escreva o valor em decimal): [tamanho]


Qual o valor do deslocamento de cada um dos campos (em bytes, escreva o valor em decimal) em relação ao início da estrutura:


a: [posicaoa]
b: [posicaob]
s.x: [posicaosx]
s.y: [posicaosy]


Complete a função C considerando o código de montagem.


```
int funcao(struct sistemascomputacao *r, int idx){
    return [ret];
}
```


```
...movl    8(%ebp), %eax
movl    12(%ebp), %edx
movswl    16(%eax,%edx,2), %eax
...
```


Resposta especificada para: tamanho  28

Resposta especificada para: posicaoa  16

Resposta especificada para: posicaob  20


Resposta especificada para: posicaosx  24

Resposta especificada para: posicaosy  28

Resposta especificada para: ret  &r->b[idx]

Respostas corretas para: ret

Método de avaliação

 Correspondência de padrão


Resposta correta

`\s*r\s*->\s*b\s*\[\s*idx\s*\]\s*`

Sensibilidade a maiúsculas e minúsculas

Respostas corretas para: posicaoa

Método de avaliação

 Correspondência de padrão

Resposta correta

`\s*0\s*`

Sensibilidade a maiúsculas e minúsculas

Respostas corretas para: posicaob		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	\s*16\s*	
Respostas corretas para: posicaosx		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	\s*20\s*	
Respostas corretas para: posicaosy		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	\s*24\s*	
Respostas corretas para: tamanho		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	\s*28\s*	

Pergunta 5



O programa em C, decParaSeq.c, foi compilado com o comando `gcc -Wall -O0 decParaSeq.c -o decParaSeq`. A figura prog+dis (ver pasta Teste 9 Junho, na secção de conteúdos da BB) mostra o programa original e o código desmontado da função `int2Seq`.

Nota: Nas respostas deverá sempre introduzir os valores em hexadecimal (prefixo **0x**), exceto quando explicitamente requerido um outro formato

a) Considere que, para efeitos de depuração, foi usado o **gdb** e que após a suspensão da execução, num determinado ponto de paragem (break), foi emitido o comando:

gdb) `printf eax=0x%x eip=0x%x ebp=0x%x , $eax, $eip, $ebp`

`eax=0x04 eip=0x80483d9 ebp=0xbfffa558`

a1. Qual o valor do registo **esp** [**esp**] ?

a2. Quando terminar a execução da função `int2Seq` qual será o valor (**em decimal**) de retorno [**reg_eax**] ?

b) No ponto de paragem, tendo em consideração o resultado da execução do comando:

gdb) `x /64xb $ebp-12`

`0xbfffa54c: 0x00 0x00 0x00 0x35 0x00 0x00 0x00 0x00`

`0xbfffa554: 0x04 0x00 0x00 0x00 0x88 0xa5 0xff 0xbf`

`0xbfffa55c: 0x26 0x84 0x04 0x08 0x2c 0x14 0x00 0x00`

`0xbfffa564: 0x88 0xa5 0xff 0xbf 0x88 0xa5 0xff 0xbf`

`0xbfffa56c: 0x86 0x84 0x04 0x08 0x00 0x00 0x00 0x00`

`0xbfffa574: 0xb8 0x3c 0x24 0x00 0xb4 0x84 0x04 0x08`

`0xbfffa57c: 0x00 0x00 0x00 0x00 0x2c 0x14 0x00 0x00`

`0xbfffa584: 0x34 0x36 0x31 0x35 0xe8 0xa5 0xff 0xbf`

b1. Em que endereços estão localizadas as variáveis locais: `nEI` [**end_nEI**], `quo` [**end_quo**] e `res` [**end_res**] ?

b2. Quais os valores das variáveis locais: (**em decimal**) `nEI` [**val_nEI**], `quo` [**val_quo**] e (**em ascii**) `res` [**val_res**] ?

c) Ainda com base na informação sobre os valores em memória obtida pelo comando `gdb` acima

c1. Qual o endereço de início da área de ativação da pilha (stack frame) [**f_main**] da função `main` ?

c2. Qual o endereço da instrução na função `main` [**reg_main**] de regresso da função `int2Seq` ?

Resposta especificada para: `esp` ☒ `0xbfffa548`

Resposta especificada para: `reg_eax` ☒ `48`

Resposta especificada para: `end_nEI` ☒ `0xbfffa54d`

Resposta especificada para: end_quo	✖ 0xbfffa54c	
Resposta especificada para: end_res	✖ 0xbfffa54c	
Resposta especificada para: val_nEl	✖ 0	
Resposta especificada para: val_quo	✔ 0	
Resposta especificada para: val_res	✖ 0	
Resposta especificada para: f_main	✖ 0xbfffa54c	
Resposta especificada para: reg_main	✖ 0xbfffa588	
Respostas corretas para: esp		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	0xbfffa548	
✔ Correspondência de padrão	\s*(?i)(0xbfffa548)\s*	
Respostas corretas para: reg_eax		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	4	
✔ Correspondência de padrão	\s*(?i)(4)\s*	
Respostas corretas para: end_nEl		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	0xbfffa554	
✔ Correspondência de padrão	\s*(?i)(0xbfffa554)\s*	
Respostas corretas para: end_quo		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	0xbfffa550	
✔ Correspondência de padrão	\s*(?i)(0xbfffa550)\s*	
Respostas corretas para: end_res		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	0xbfffa54f	
✔ Correspondência de padrão	\s*(?i)(0xbfffa54f)\s*	
Respostas corretas para: val_nEl		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	4	
✔ Correspondência de padrão	\s*(?i)(4)\s*	
Respostas corretas para: val_quo		
Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ Correspondência de padrão	0	
✔ Correspondência de padrão	\s*(?i)(0)\s*	

Respostas corretas para: val_res

Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	5	
✔ <i>Correspondência de padrão</i>	\s*(?i)(5)\s*	

Respostas corretas para: f_main

Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	0xbfffa588	
✔ <i>Correspondência de padrão</i>	\s*(?i)(0xbfffa588)\s*	

Respostas corretas para: reg_main

Método de avaliação	Resposta correta	Sensibilidade a maiúsculas e minúsculas
✔ <i>Correspondência de padrão</i>	0x8048426	
✔ <i>Correspondência de padrão</i>	\s*(?i)(0x8048426)\s*	