| Nº 92839 | Nome: BEATRIZ SOUSA DEMÉTRIO | Turma: PL2 |
|---|---|---|

## Resolução dos exercícios (deve ser redigido manualmente)

### 1. Código C de um `main` simples que invoque a função `getline`

Copie para aqui o código C de um `main` simples que colocou no servidor remoto (para invocar a função `getline`).

```c
#include <stdio.h>

int main()
{
    char *x = getline();
    printf("%s\n", x);
    return 0;
}
```

### 2. Análise do código desmontado

Compile o código C sem qualquer otimização (com `-O0`) e copie para aqui o código executável "desmontado" (*disassembled*) da função `getline` até à chamada da função `gets`, mostrando (com um *print screen* ou foto do monitor) todos os comandos que usou para compilar e para ter o código desmontado da função.
Anote detalhadamente o código desmontado, ignorando as fases de arranque e término da função.

```
[03a92839@sc tpc8]$ gcc -Wall -O0 getline.c -o getline
/tmp/ccKvg95h.o(.text+0xe): In function `getline':
: the `gets' function is dangerous and should not be used.
[03a92839@sc tpc8]$ gdb getline
GNU gdb Red Hat Linux (6.3.0.0-1.138.el3rh)
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "i386-redhat-linux-gnu"...(no debugging symbols found)
Using host libthread_db library "/lib/tls/libthread_db.so.1".

(gdb) disass getline
```

```
0x08048434 <getline+0>: push   %ebp                    #começa a função
0x08048435 <getline+1>: mov    %esp,%ebp               # ebp=esp
0x08048437 <getline+3>: sub    $0x18,%esp              #sobe o topo da stack 24 células
0x0804843a <getline+6>: sub    $0xc,%esp               #sobe o topo da stack 12 células
0x0804843d <getline+9>: lea    0xfffffff8(%ebp),%eax   #guardar em %eax o valor contido no registo ebp subtraindo lhe 8 em complemento para 2
0x08048440 <getline+12>:        push   %eax            # reserva espaço no topo da stack (decrementa o %esp) e escreve no topo o conteúdo de %eax
0x08048441 <getline+13>:        call   0x8048328       # invoca gets
```

### 3. Execução do código

**Replique** aqui tudo que apareceu no monitor assim que mandou executar o código (incluindo os caracteres que tiver introduzido e o resultado da execução do código).

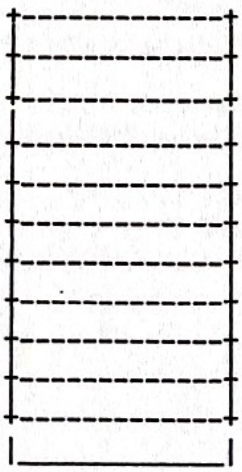
```
(gdb) run
Starting program: /home/03a92839/tpc8/getline
(no debugging symbols found)
(no debugging symbols found)
1234567890123

Program received signal SIGSEGV, Segmentation fault.
0x080483c8 in call_gmon_start ()
```

### 4. Estimando o quadro da função `getline` na *stack*

A anomalia que constatou poderá ser devida a um erro na função `getline` que provocou a alteração indevida de informação armazenada na *stack*. Isso pode conduzir a que a execução do código tenha tentado aceder a uma zona de memória que não faz parte da área de memória que estava alocada a este programa. Tal pode acontecer no regresso de uma função, se o valor do endereço de regresso (que está na *stack*) tiver sido indevidamente modificado.
Para verificar se foi isto que aconteceu, temos de analisar o quadro da função `getline`.

**Preencha** o diagrama do quadro da função `getline` (a sua *stack frame*) que é gerado até este ponto da execução (até à linha 5), com a estimativa dos seus valores e endereços, procedendo assim:

a. **Indique** à esquerda das caixas (cada caixa representa 4 células de memória) a posição apontada pelo registo `%esp` e pelo registo `%ebp`, bem como outras posições relevantes (nesse caso, utilize posições relativas ao registo `%ebp`, e.g., `%ebp+4`, etc.);
b. **coloque** à direita de cada caixa uma etiqueta que descreva o que representa a caixa (nota: algumas caixas podem conter valores irrelevantes);
c. **coloque** dentro da caixa correspondente o endereço de regresso para a `main` (nota: consulte o código desmontado da função `main`)



**endereço de regresso para a *main***