Esercizio su buffer overflow:
il codice seguente riporta un buffer di 10 caratteri

```
  GNU nano 7.2                                  BOF.c *
#include <stdio.h>

int main () {

char buffer [10];

printf ("Si prega di inserire il nome utente:");
scanf ("%s", buffer );

printf ("Nome utente inserito; %s\n", buffer);

return 0;

}
```

Inserendo un nome utente da 10 caratteri il programma non presenta errori.
Inserendo invece un input maggiore,ad esempio di 30,il programma risponde con un “segmentation fault”.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nano BOF.c

┌──(kali㉿kali)-[~/Desktop]
└─$ gcc -g BOF.c -o BOF

┌──(kali㉿kali)-[~/Desktop]
└─$ ./BOF
Si prega di inserire il nome utente:1234567890
Nome utente inserito; 1234567890

┌──(kali㉿kali)-[~/Desktop]
└─$ ./BOF
Si prega di inserire il nome utente:qwertyuiopasdfghjklzxcvbnmqwer
Nome utente inserito; qwertyuiopasdfghjklzxcvbnmqwer
zsh: segmentation fault  ./BOF
```

Dei seguito il codice riportato con buffer “30”:

```c
#include <stdio.h>

int main () {

char buffer [30];

printf ("Si prega di inserire il nome utente:");
scanf ("%s", buffer );

printf ("Nome utente inserito; %s\n", buffer);

return 0;

}
```

Inserendo 30 caratteri cone nome utente il programma non presenta errori.
Inserendo più di 30 caratteri,ad esempio di 55,il programam risponde con il medesimo errore “segmentation fault”.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo nano BOF.c

┌──(kali㉿kali)-[~/Desktop]
└─$ gcc -g BOF.c -o BOF

┌──(kali㉿kali)-[~/Desktop]
└─$ ./BOF
Si prega di inserire il nome utente:qwertyuiopasdfghjklzxcvbnmqwer
Nome utente inserito; qwertyuiopasdfghjklzxcvbnmqwer

┌──(kali㉿kali)-[~/Desktop]
└─$ ./BOF
Si prega di inserire il nome utente:qwertyuiopasdfghjklzxcvbnm1234567890qwertyuiopasdfghjkl
Nome utente inserito; qwertyuiopasdfghjklzxcvbnm1234567890qwertyuiopasdfghjkl
zsh: segmentation fault  ./BOF

┌──(kali㉿kali)-[~/Desktop]
└─$
```