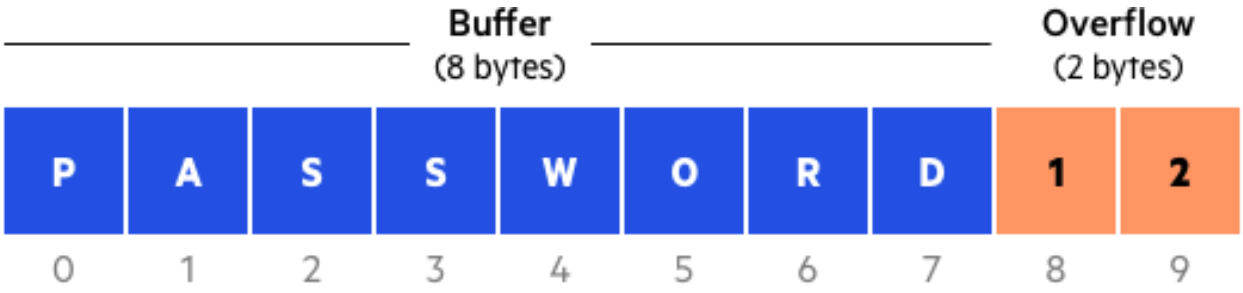
# Esercizio S7-L4

## [1] Traccia dell'esercizio:

L'esercizio di oggi, ci chiede di effettuare un BoF (Buffer Overflow) su un programma compilato in C con un array di caratteri avente dimensione 30.

## [2] Definizione di buffer overflow:

Il buffer overflow (trabbocamento della memoria tampone), è una condizione che si verifica quando in un'area di memoria definita, si inseriscono più dati di quanti previsti.




## [3] Svolgimento del esercizio:

Come prima cosa scriviamo il programma in C utilizzando l'editor di testo nano.

```c
#include <stdio.h>

int main()
{

//Dichiarazione del buffer di tipo char [Char Array = String]
char buffer [30];


//Interazione con l'utente
printf("Inserisci il tuo nome utente: ");
scanf("%s", buffer);

printf("Il nome utente da lei inserito è: %s\n", buffer);


//Ritorna il codice "0" se tutto è corretto
return 0;

}
```

Una volta scritto il programma lo compiliamo con il comando gcc (GNU Compiler collection) e lo eseguiamo con ./<nome del programma> .

```
C:\home\kali\esC> gcc -g BoF.c -o BoFCompiled

C:\home\kali\esC> ./BoFCompiled
Inserisci il tuo nome utente: abcdefghilmnopqrstuvzabcdefghilmnopqrstuvz
Il nome utente da lei inserito è: abcdefghilmnopqrstuvzabcdefghilmnopqrstuvz
zsh: segmentation fault  ./BoFCompiled

C:\home\kali\esC> ./BoFCompiled
Inserisci il tuo nome utente: Francesco
Il nome utente da lei inserito è: Francesco
```

## [4] Conclusioni:

Come notiamo nel primo caso inserendo più di 30 caratteri si verifica un buffer overflow. Ricordiamo che il buffer overflow può portare al crash del programma o addirittura alla manipolazione del codice.