

Para cada uma das funções abaixo, cite qual a sua assinatura de função, explicando seus parâmetros de entrada e de retorno. Por fim, exemplifique o uso dessas funções com pequenos trechos de código simples

---

### a) strlen

#### Assinatura

Recebe uma string e devolve o seu comprimento

```
unsigned int strlen (char *s)
```

#### Parâmetros

- **\*s**: String que terá o comprimento calculado

#### Exemplo

```
#include <stdio.h>
#include <string.h>

int main(){
    int len;
    char array[20]="ifb.edu.br";

    len = strlen(array);

    printf( "\nstring length: %d \n" , len );
    return 0;
}
```

### b) strcmp

#### Assinatura

Recebe duas strings, e compara lexicograficamente retornando:

- **zero**: se as duas string forem **iguais**
- **um valor > 0**: se **\*s** for lexicograficamente maior do que **\*t**
- **um valor < 0**: se **\*s** for lexicograficamente menor do que **\*t**

```
int strcmp (char *s, char *t)
```

### Parâmetros

- **\*s**: primeira String da comparação
- **\*t**: segunda String da comparação

### Exemplo

```
#include <stdio.h>
#include <string.h>
int main() {
    char str1[] = "if";
    char str2[ ] = "ifb";
    int i, j, k;

    i = strcmp ( str1, "if" );
    j = strcmp ( str1, str2 );
    k = strcmp ( str1, "f" );

    printf ( "\n%d %d %d", i, j, k );
    return 0;
}
```

## c) strcpy

### Assinatura

Recebe duas strings e copia o conteúdo da segunda para a primeira, incluindo o byte nulo final

```
void strcpy (char *s, char *t)
```

### Parâmetros

- **\*s**: string que receberá o conteúdo adicional
- **\*t**: string contendo o conteúdo adicional

### Exemplo

```
#include <stdio.h>
#include <string.h>

int main() {
    char source[] = "ifb.edu.br";
```

```
char target[20]= "";

printf ( "\nsource string = %s", source );
printf ( "\ntarget string = %s", target );

strcpy ( target, source );

printf ( "\ntarget string after strcpy( ) = %s", target
);
return 0;
}
```

## d) strcat

### Assinatura

Concatena duas strings passadas por parâmetro. **source** é concatenado no final de **destination**.

```
char * strcat ( char *destination, const char *source
);
```

### Parâmetros

- **\*destination:** String que receberá o conteúdo
- **\*source:** nova parte

### Exemplo

```
#include <stdio.h>
#include <string.h>

int main(){
    char source[] = " ifb.edu.br";
    char target[]= " atividade de fixação 2";

    printf ( "\nSource string = %s", source );
    printf ( "\nTarget string = %s", target );

    strcat ( target, source ) ;

    printf ( "\nTarget string after strcat( ) = %s", target
);
}
```

## e) atoi

## Assinatura

Converte uma variável do tipo `string` para `int`

```
int atoi (const char *str);
```

## Parâmetros

- **\*str**: Valor que será convertido de string para int

## Exemplo

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    char a[10] = "100";
    int value = atoi(a);
    printf("Value = %d\n", value);
    return 0;
}
```