Programación de software de sistemas

Ayudantía 2

Profesor: Rodrigo Verschae

Ayudante: Nicolás Araya

P1 string_len

Crear una función “int string_len(char *str)” que retorne la cantidad de char legibles en un string.

```
char *str = “Hello, world!”;
int len = 0;
```

‘H’ ‘e’ ‘l’ ‘l’ ‘o’ ‘,’ ‘ ‘ ‘w’ ‘o’ ‘r’ ‘l’ ‘d’ ‘!’ ‘\0’

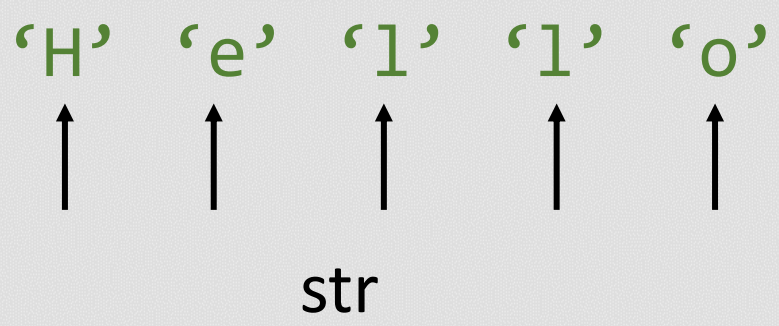
Solución

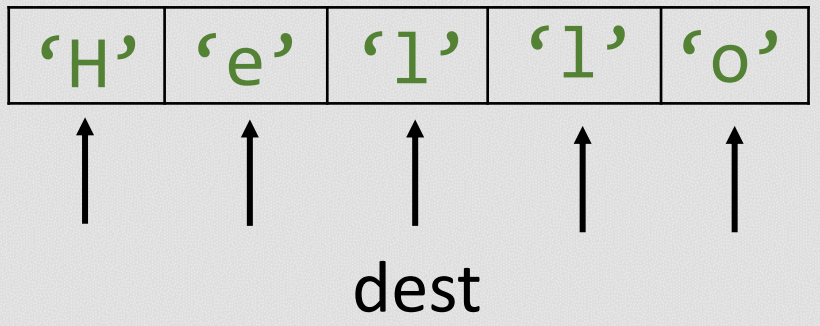
```
int string_len(char *str)
{
      int len = 0;
      while( *str != '\0' )
      {
            len++;
            str++;
      }
      return len;
}
```

P2 copy_string

Crear función “void copy_string(char *str, char *dest)” que copie los char de un string a otro char inicializado en main.

```
char *str = “Hello”;
char *dest[str_len(str)];
```





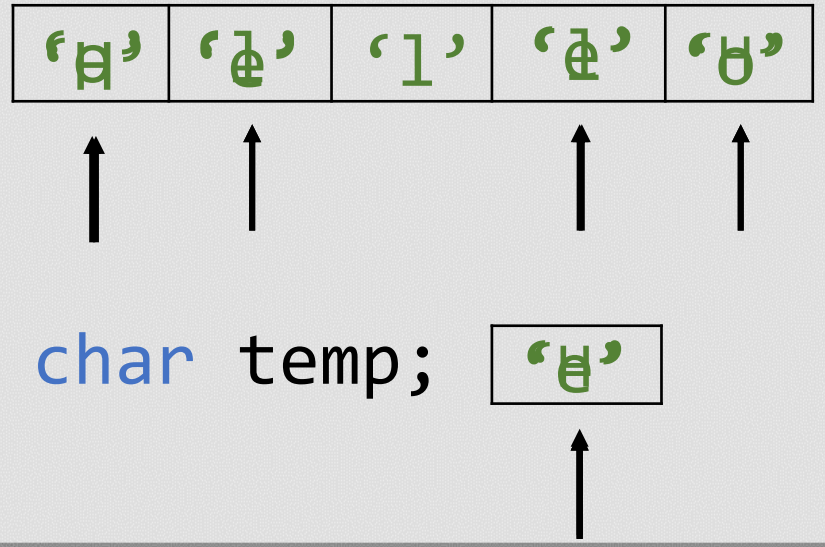
Solución

```
void copy_string(char *str, char *dest)
{
    while( *str != '\0' )
    {
        *dest++ = *str++;
    }
    *dest = '\0';
}
```

P3 reverse_string

Crear función “void reverse_string(char *str)” que reordene en reversa los char del string str.

```
char *str = “Hello”;
```




```
char temp;
```

Solución

```
Void reverse_string(char *str)
{
        char aux;
        int len = string_len(str);
        int j = len - 1;

        for (int i = 0; i < len / 2 ; i++)
        {
                aux = str[i];
                str[i] = str[j];
                str[j] = aux;
                j--;
        }
}
```

P4 reverse_string

Cree la función “char *reverse_string(char *str)” que retorne una nueva cadena de char usando una variable dinámica y la asigne a una variable en main.

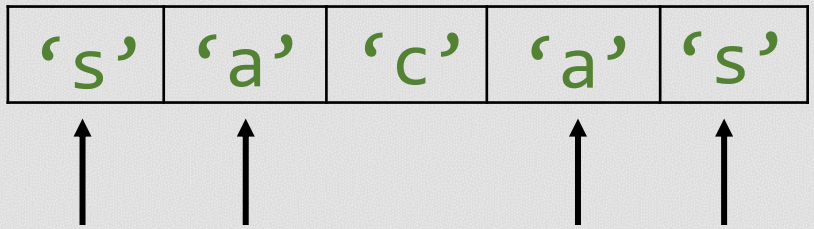
Solución

```
char *reverse_string(char *str)
{
        int len = string_len(str);
        char *ans = malloc(sizeof(char)*len);

        int j = len - 1;

        while (*str != '\0')
        {
                ans[j] = *str++;
                j--;
        }

        return ans;
}
```

P5 Palíndromo

Cree la función “int palindromo(char *str)” que retorne 1 si es palíndromo o 0 si no lo es.

```
char *str = “sacas”;
```




```
char *j = str;
char *i = ;
```

Solución

```c
char *reverse_string(char *str)
{
        char *i = str;
        char *j = str + str_len(str) - 1;
        while ( i < j )
        {
                if(*i != *j){
                        return 0;
                }
                i++;
                j--;
        }
        return 1;
}
```

```c
char *reverse_string(char *str)
{
        int j = str_len(str) -1;

        for ( int i = 0;i < str_len(str);i++ )
        {
                if ( str[i] != str[j] ){
                        return 0;
                }
                j--;
        }
        return 1;
}
```

ayudantía! [+] 1,0-1 Top

Consultas!