**2017-08-16, rgf**

<https://stackoverflow.com/questions/9834067/difference-between-char-and-const-char>

* const char\*  
  is a **mutable** pointer to an **immutable** character/string. You cannot change the contents of the location(s) this pointer points to. Also, compilers are required to give error messages when you try to do so. For the same reason, conversion from const char \* to char\* is deprecated.
* char\* const  
  is an **immutable** pointer (it cannot point to any other location) **but** the contents of location at which it points are **mutable**.
* const char\* const  
  is an **immutable** pointer to an **immutable** character/string.
* char\*  
  is entirely permissive.

**rexx.h**

typedef const char \* CSTRING /\* pointer to zero-terminated (read-only) string \*/

typedef struct \_CONSTRXSTRING { /\* const rxstr \*/

size\_t strlength; /\* length of string \*/

const char \*strptr; /\* pointer to string \*/

} CONSTRXSTRING;