```
//*****************************************************************************
//*****************************************************************************
//                                                **
//                       Operacje proste na łańcuchach znakowych    **
//      funkcjonalnosci:                                        **
//                    - Kopiowanie jednego łańcucha znaków do drugiego           **
//                    - Porównywanie dwóch łańcuchów (równe czy różne)          **
//                    - Dopisywanie jednego łańcucha na koniec drugiego         **
//                    - Zamienianie znaków w łańcuchu                    **
//                                                      **
//*****************************************************************************


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

#define NULL '\0'

#define NOTEQUAL DIFFERENT
#define EQUEL EQUAL
enum CompResult {DIFFERENT, EQUAL};



void CopyString(char pcSource[], char pcDestination[]){
    for (unsigned char ucStrIndex = 0; NULL != pcSource[ucStrIndex]; ucStrIndex++) {
        pcDestination[ucStrIndex] = pcSource[ucStrIndex];
        pcDestination[ucStrIndex + 1] = NULL;
    }
}



enum CompResult eCompareString(char pcStr1[], char pcStr2[]){
    for (unsigned char ucStrIndex = 0; NULL != pcStr1[ucStrIndex] || pcStr2[ucStrIndex] != NULL; ucStrIndex++) {
        if(pcStr1[ucStrIndex] != pcStr2[ucStrIndex]){
            return NOTEQUAL;
        }
    }

    return EQUEL;
}
```

```
void AppendString(char pcSourceStr[], char pcDestinationStr[]) {
    unsigned char ucDestLastIndex;

    for (ucDestLastIndex = 0; NULL != pcDestinationStr[ucDestLastIndex]; ucDestLastIndex++) {};

    CopyString(pcSourceStr, &pcDestinationStr[ucDestLastIndex]);
}



void ReplaceCharactersInString(char pcString[],char cOldChar,char cNewChar) {
    for (unsigned char ucCharIndex = 0; NULL != pcString[ucCharIndex]; ucCharIndex++) {
        if(pcString[ucCharIndex] == cOldChar){
            pcString[ucCharIndex] = cNewChar;
        }
    }
}
```