Hint. Don't forget the all of the strings has a hidden char = 0 or '\0' and the end.

1. Suppose the function "fun" has two inputs string s ( char \*) and char c, the function will search in s to find if there is a char having same value as c. if found, insert c again after this char.

If input s = "bbaacda" c = 'a' . then result s = "bbaaaacdaa"

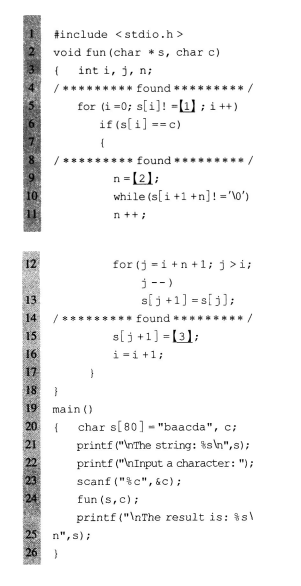
try to find what to put in the blank [1] [2] and [3]

[1]: ‘\0’

[2]: ‘0’

[3]: ‘c’

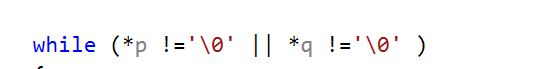
any memory that allocated to you is random. if you don't do initialize. anything is random.

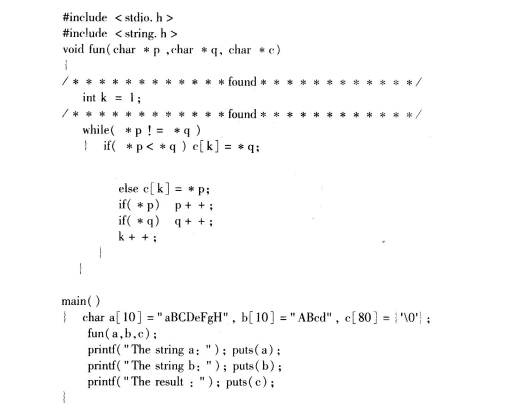


2. The function "fun" will try to compare the two strings in p and q from the first position to the last. The comparing criteria is to find the char which has larger ASCII value. It will put the comparision result in string c. If p and q don't have the same length, then only compare the part with same length, and put the rest of the longer string directly to c.

If p = "aBCDeFgH" q = "Abcd" . then c = "abcdeFgH" because 'a' > 'A', 'b' > 'B', 'c' > 'C', 'd' > 'D'

There are two line of code have bugs. please find them and correct them.

1   
  
C:\Users\Yu\Documents\Tencent Files\651412330\Image\C2C\`MILZW444]ZCOU{A37{KGA3.png  
  
  
  




char a[10] = "aBC", b[10] = "ABcdaaaaa", c[80] = {'a', '\0' };

fun(a, b, c);

printf("The string a: "); puts(a);

printf("The string b: "); puts(b);

printf("The string c: "); puts(c);

void fun(char \*p, char \*q, char \*c)

{

int k = 0;

while (\*p !='\0' || \*q !='\0' )

{

if (\*p < \*q)

{

c[k] = \*q;

}

else

{

c[k] = \*p;

}

k++;

if (\*p)

{

p++;

}

if (\*q)

{

q++;

}

}

}