mergeStr

Write a C function mergeStr(), which takes in 3 parameters ***a***, ***b*** and ***c***, merges two alphabetically ordered character strings ***a*** and ***b*** into character string ***c*** according to alphabetical order. For example, if ***a*** is "agikmpq" and ***b*** is "bcdefhjlnr", then the resultant string ***c*** will be "abcdefghijklmnpqr". The string ***c*** will be passed to the caller via call by reference.

A sample program template is given below:

```
#include <stdio.h>
#include <string.h>
void mergeStr(char *a, char *b, char *c);
int main()
{
   char a[80],b[80];
   char c[80];

   printf("Enter the first string a: \n");
   scanf("%s",a);
   printf("Enter the second string b: \n");
   scanf("%s",b);
   mergeStr(a,b,c);
   printf("mergeStr(): %s", c);
   return 0;
}
void mergeStr(char *a, char *b, char *c)
{
   /* Write your code here */
}
```

Some test input and output sessions are given below:

(1) Test Case 1:

```
Enter the first string a:
ace
Enter the second string b:
bdg
mergeStr(): abcdeg
```

(2) Test Case 2:

```
Enter the first string a:
agikmpq
Enter the second string b:
bcdefhjlnr
mergeStr(): abcdefghijklmnpqr
```

(3) Test Case 3:

```
Enter the first string a:
afk
Enter the second string b:
afk
mergeStr(): aaffkk
```

(4) Test Case 4:

```
Enter the first string a:
```

```
afkm
Enter the second string b:
bbbggg
mergeStr(): abbbfgggkm
```

```
#include <stdio.h>
#include <string.h>
void mergeStr(char *a, char *b, char *c);
int main()
{
   char a[40],b[40];
   char c[40],*p;

   printf("Enter the first string: \n");
   fgets(a, 40, stdin);
   if (p=strchr(a,'\n')) *p = '\0';
   printf("Enter the second string: \n");
   fgets(b, 40, stdin);
   if (p=strchr(b,'\n')) *p = '\0';
   mergeStr(a,b,c);
   printf("mergeStr(): ");
   puts(c);
   return 0;
}
void mergeStr(char *a, char *b, char *c)
{
    char k,*p,*q,*r;

    strcpy(c,strcat(a,b));

    for(p=c;*p;p++)
    {
        for(q=r=p;*q;q++)
        {
            if(*r>*q)
            {
                r=q;
            }
        }
        if(r!=p)
        {
            k=*r;
            *r=*p;
            *p=k;
        }
    }
}
```

sort function