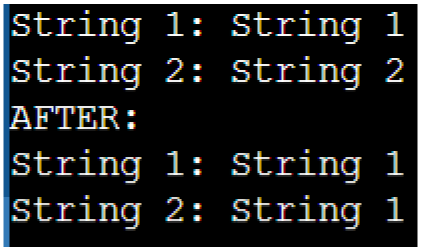
Part 1

```
#include <stdio.h>

char* strcpy(char *strDest, const char *strSrc) {
     // Set the destination string to the pointer copied so the
address is not lost
     char *copied = strDest;
     // Copies the sorce string to destination string character by
chracter
     while(*strSrc != '\0')
     {
     *strDest = *strSrc;
     strDest++;
     strSrc++;
     }
     // Returns string pointing to the destination string
     return copied;
}

int main() {
     char str1[] = "String 1";
     char str2[] = "String 2";
     printf("BEFORE:\nString 1: %s\nString 2: %s\n",str1,str2);
     strcpy(str2,str1);
     printf("AFTER:\nString 1: %s\nString 2: %s",str1,str2);
     return 0;
}
```

1. 

```
String 1: String 1
String 2: String 2
AFTER:
String 1: String 1
String 2: String 1
```

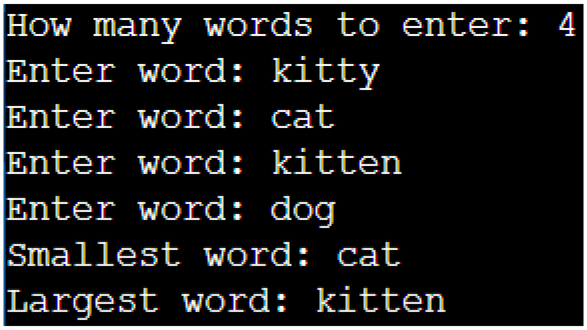
2. It returns a pointer so that the whole string can be returned and not just the first character

Part 2

```
#include <stdio.h>
#include <string.h>

int main() {
     // initializing variables
     char smallest_word[20] = "12345678901234567890";
     char largest_word[20] = "1";
     int num_of_words;
     // Asking user how many words are in their list
     printf("How many words to enter: ");
     scanf("%d", &n);
     // Consumes newline character left in buffer from scanf
     getchar();
     // For loop runs until it reaches the number of words
     for(int i = 1; i <= num_of_words; i++){
     // Word string resets with every iteration
     printf("Enter word: ");
     char word[20];
     fgets(word,20,stdin);
     // Remove newline character left from fgets
     word[strlen(word)-1] = '\0';
     // Comparing entered word length to current minimum and current
maximum
     // Swap max or min if word is smaller or larger
     if(strlen(word)-1 < strlen(smallest_word)-1) {
          strcpy(smallest_word,word);
     }
     if(strlen(word)-1 > strlen(largest_word)-1) {
          strcpy(largest_word,word);
     }
     }
     // Print results
     printf("Smallest word: %s\nLargest word:
%s",smallest_word,largest_word);
     return 0;
}
```

3.



```
How many words to enter: 4
Enter word: kitty
Enter word: cat
Enter word: kitten
Enter word: dog
Smallest word: cat
Largest word: kitten
```