Computation I 5EIA0
Homework 2: Arrays and Strings (v4.3, September 12, 2022)
Deadline Tuesday 20 September 13:30

Task 1. Create a C file and declare declare the following variables in your main function:

```
char string1[12] = "hello";
char string2[12] = "hello\n";
char string3[12] = "hello\n\0you\n";
char string4[12] = "0123456789AB";
```

The first function that we will write is the `void printArray(char a[], int length)` function. It has an array of characters as input and an integer that indicates the length of the array. `void` signifies that it has no return value. The function prints the `'['` character and then all characters in the array, separated by commas, finally followed by the `']'` and `'\n'` characters. When a character in the string is null then you should print `\0`, i.e. the backslash character followed by `'0'`. When a character in the string is the newline character then you should print `\n`, i.e. the backslash character followed by `'n'`.
Call `printArray` with each of the string variables as input. The output should be as follows:

```
[h,e,l,l,o,\0,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,y,o,u,\n,\0]
[0,1,2,3,4,5,6,7,8,9,A,B]
```

Can you explain the output? (Hint: how are arrays initialised?)

Task 2. The second function that we will write is the `void printString(char str[])` function. It has a single input, which is an array of characters. A string is an array of characters that is terminated by the null character `'\0'`. We could just use the `printf` function in the `stdio` library, but that would be too easy. Instead we will use the `putchar()` function to print a single character at a time. Write the function `printString` and call it on the four character arrays. The output should be something like this:

```
[h,e,l,l,o,\0,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,y,o,u,\n,\0]
[0,1,2,3,4,5,6,7,8,9,A,B]
"hello"
"hello
"
"hello
"
"0123456789ABhello
"
```

Can you explain what causes the double quote of the `printString(string2);` command to be printed on a new line? Can you explain why `you` is not printed in the `printString(string3);` command? And what about the final `printString(string4);` command? (Your output may be something different like `"0123456789AB????"`.)

Hint: See the explanation at the end of this document.

Task 3. Next, let's read input from the terminal using the `getchar()` function. Write a `void readLine(char str[], int length)` function that reads a character from the terminal until it has read a newline character. (See the lecture notes on how to do this!) All characters up to (i.e. excluding) the newline character are placed in the `str` character array. Then terminate the string (in the `str` array) with a null character. You can use the following main program to get started.

```
#define LENGTH 30
int main(void)
{
  ...
  while (1) {
    printf("String? ");
    readLine(inputString, LENGTH);
    printArray(inputString, LENGTH);
    printString(inputString);
  }
}
```

If there is not enough space in the array then store the first `length-1` characters and the null character, but keep reading from the terminal until the newline. Some of the characters won't be in the string, but the resulting string must always be null terminated.

Now that you are familiar with the difference between characters, strings, and arrrays it is time to start the development of our program. As a first step, you will now develop several functions that are useful for our program. To test the correctness of each of these function you will have to modify the `main` function each time you complete a task. Make sure that each function is correct before proceeding to the next one, otherwise it is much harder to debug the program as a whole. You will write an interactive program that you can give commands. These are the commands that your program will support.

command	operation
q	quit program
s	enter a string
p	print string using printString
a	print string using printArray
o	find first occurence of character in string
i	insert a character in string
r	replace characters in string
R	reorder string

function	1	2	3	4	5	6	7	8	9	10	11	12	13	% per fn	cumulative %
quit	1	1	1	1	1	1	1	1	1	1	1	1	1	8%	8%
string		1	1	1	1	1	1	1	1	1	1	1	1	8%	15%
print string			1	1	1	1	1	1	1	1	1	1	1	8%	23%
print array				1	1	1	1							8%	31%
occurrence					1	1	1							15%	46%
insert							1	1						15%	62%
replace									1	1				15%	77%
reorder											1	1	1	23%	100%

Figure 1: Test cases.

Task 4. Modify the main function to ask for a single-character command with the `readLine` function. Print the error message shown below if an invalid command is given. Your code will look something like this:

```
do {
  printf("Command? ");
  readLine(...);
  switch (inputString[0]) {
  case 'q':
    printf("Bye!\n");
    break;
  default:
    printf("Unknown command '%c'\n",inputString[0]);
    break;
  }
} while (inputString[0] != 'q');
```

This would be a possible output:

```
Command? x
Unknown command 'x'
Command?    xyz
Unknown command ' '
Command? quit
Bye!
```

The second input was three spaces, followed by xyz. Note that trailing characters (`xyz` and `uit`) are ignored. (See the screencasts on Oncourse for more details on `getchar` and `scanf`.)

Task 5. Define a string `currString` of length LENGTH in the main function. Implement the 'p' command to print the string using the `printString` function, and the 'a' command to print the string using the `printArray` function.

```
Command? p
The current string is: ""
Command? a
The current array is: [\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
Command? q
Bye!
```

Hint: ensure that your arrays are initialised.

Task 6. Add the 's' command to ask for a new value for the `currString` string using your `readLine` function.

```
Command? s
Please enter a string?   Bert and Ernie
Command? p
The current string is: "  Bert and Ernie  "
Command? a
The current array is: [ , ,B,e,r,t, ,a,n,d, ,E,r,n,i,e, , ,\0,\0,\0,\0,\0,\0,\0,\0,\0,
Command? q
Bye!
```

Task 7. Write a function `int findFirstOccurrence(char str[], char aChar)` that searches for a specific character `aChar` in the string. The return value of this function is the index in `str` of the first occurrence of the character `aChar`. When `aChar` is not found, the value `-1` is to be returned. Add the 'o' (occurrence) command, and read the necessary function arguments from the keyboard using your `readLine` function.

```
Command? s
Please enter a string? whatever
Command? o
Find first occurrence of which character? a
The first occurrence of 'a' is at index 2
Command? o
Find first occurrence of which  character? e
The first occurrence of 'e' is at index 4
Command? o
Find first occurrence of which  character? x
The first occurrence of 'x' is at index -1
```

Hint: Although you only need a single character `aChar` you can use `readLine` to read an entire line and then just use the first character.

Task 8. Write a `int readInt(char str[], int length)` function that returns the positive integer of the number in the string. e.g. when str is "0123" then the function should return 123. You only need to deal with positive integers, i.e. at least zero. Stop at the first non-digit. If there is no integer (e.g. the character at index start is non-digit character) then return -1. Some examples

```
str == "123"   --> return 123
str == "0123 " --> return 123
str == "0123x" --> return 123 (stop at the x)
str == " 123"  --> return -1 (stop at the leading space)
str == "x"     --> return -1 (stop at the x)
str == ""      --> return -1 (empty string)
```

Hint: You need a loop in which you check if the current character is a digit (character `'0'` to `'9'`). If not, then exit the loop. If yes, then add its decimal value to the current integer value.

Task 9. Add the command 'i' (insert) that uses a new function `insertChar(char str[], char aChar, int index)` that inserts character `aChar` at position `index` in the string `str`. Use your `readLine` function to read `aChar` from the terminal. Then use your `readLine` and `readInt` functions to read an integer from the terminal.

```
Command? s
Please enter a string? I like writing C code
Command? i
Insert which character? d
At what index? 6
Command? p
The current string is: "I liked writing C code"
Command? i
Insert which character? s
At what index? 22
Command? p
The current string is: "I liked writing C codes"
Command? q
Bye!
```

Hint: Always ensure that you do not crash the program. If the index is larger than the length of the string, or if inserting the character would overflow the array, then do not insert anything.

Task 10. Write a function `replaceChars(char str[], char fromString[], char toChar)` that replaces all occurrences of the characters in the array `fromString` in the string `str` by the character `toChar`. The return value is the number of replaced characters. Add the 'r' (replace) command and read the strings `fromString` and `toChar` from the keyboard using your `readLine` function. Call `replaceChars` with `fromString` and `toChar[0]` as input.

```
case 'r':
  printf("Replace which characters? ");
  // read fromString
  printf("with which character? ");
  // read toChar
  replaceChars(currString, fromString, toChar[0]);
  break;
```

```
Command? s
Please enter a string? I like writing C code
Command? p
The current string is: "I like writing C code"
Command? r
Replace which characters? i
with which character? i
Command? p
The current string is: "I like writing C code"
Command? r
Replace which characters? ioe
with which character? -
Command? p
The current string is: "I l-k- wr-t-ng C c-d-"
Command? q
Bye!
```

```
(gdb) run
Starting program: /home/computation/a.out
Command [qpasoirR123]? s
Please enter a string? I like debugging C code
Command [qpasoirR123]? p
The current string is: "I like debugging C code"
Command [qpasoirR123]? r
Replace which characters? i
with which character? i
^C
Program received signal SIGINT, Interrupt.
__strlen_avx2 () at ../sysdeps/x86_64/multiarch/strlen-avx2.S:61
61      ../sysdeps/x86_64/multiarch/strlen-avx2.S: No such file or directory.
(gdb) where
#0  __strlen_avx2 () at ../sysdeps/x86_64/multiarch/strlen-avx2.S:61
#1  0x0000555555554b0f in findFirstOccurrence (
    str=0x7fffffffdce0 "I like debugging C code", aChar=105 'i')
    at doubledutch.c:73
#2  0x0000555555554bf2 in replaceChars (
    str=0x7fffffffdce0 "I like debugging C code", aChars=0x7fffffffdd00 "i",
    rChar=105 'i') at doubledutch.c:94
#3  0x0000555555555150 in main () at doubledutch.c:199
(gdb) quit
A debugging session is active.

        Inferior 1 [process 9793] will be killed.

Quit anyway? (y or n) y
computation@computation-virtual-machine:~$
```

Figure 2: Be careful when replacing a character with itself! If your program doesn't terminate then use the gdb debugger. Interrupting the program with control-C will then tell you exactly where the problem is. Here the output shows the stack, i.e. main called replaceChars called findFirstOcurrence.

Task 11. Write a function `void stringReorder(char str[], int index1, int index2)` that divides a string in three parts, and puts them together in a different way. The string `str` is cut at the positions `index1` and `index2`. After splitting the string into substring1, substring2, substring3 they are reordered to substring3, substring2, substring1. Add the 'R' (reorder) command to your main loop, and read `str`, `index1` and `index2` from the keyboard, such that the output looks like this:

```
Command? s
Please enter a string? Ernie and Bert
Command? R
Please enter index 1? 5
Please enter index 2? 10
Command? p
The current string is "Bert and Ernie"
Command? q
```

In this example the three parts are "Ernie " (notice the trailing space), "and " (notice the trailing space), and "Bert".

See hints on the next page...

Hint: It is really helpful to solve this on paper first! Draw the string with its three substrings and their starting indices on paper, before and after the reordering.

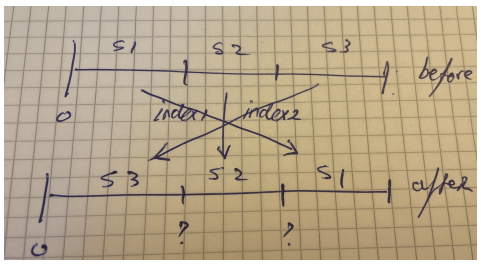

Figure 3: Visualing the reorder function to determine the right indices to copy correctly.

Given the two indices supplied by the user you need to compute the two new indices indicates by the question marks in the photo. (Do this in general: given index i1 and index i2, new index j1 is some expression involving the i1, i2, and the length of the string. Similarly for new index j2.) After that you you need to copy three substrings into a temporary array, which you can do with the `strncpy` function from the `string.h` library. (Type `man strncpy` in the terminal or look it up in K&R. It is useful to know these functions because you can use them in the exam.) Finally, copy the temporary array back into the original array. (You have to use a temporary array because the regions that you're copying may be overlapping and `strncpy` doesn't support that.) Also, when you developing the program it is helpful to print out what you're doing, e.g.

```
Command? s
Please enter a string? 123 456 789
Command? R
Please enter index 1? 3
Please enter index 2? 8
copy 8-11 to 0
copy 3-8 to 3
copy 0-3 to 8
Command? p
The current string is: "789 456 123"
```

Some useful test cases:


```
Command? s
Please enter a string? 0123456789
Command? R
Please enter index 1? 0
Please enter index 2? 4
Command? p
The current string is: "4567890123"
Command? s
Please enter a string? 0123456789
Command? R
Please enter index 1? 1
Please enter index 2? 2
Command? p
The current string is: "2345678910"
Command? s
Please enter a string? 0123456789
Command? R
Please enter index 1? 0
Please enter index 2? 10
Command? p
The current string is: "0123456789"
Command? q
Bye!
```

Hint: Explanation of `printString` output:

```
[h,e,l,l,o,\0,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,\0,\0,\0,\0,\0]
[h,e,l,l,o,\n,\0,y,o,u,\n,\0]
[0,1,2,3,4,5,6,7,8,9,A,B]
"hello"
"hello
"
"hello
"
"0123456789ABhello
"
```

- Can you explain what causes the double quote of the `printString(string2);` command to be printed on a new line? The string contains the newline character, which is printed out before final double quote.
- Can you explain why you is not printed in the `printString(string3);` command? A string finishes with a null character (`'\0'`). This is why when we pass a string to a function we do not indicate the length of the array in which the string is stored. So our printString function correctly stops printing when it encounters a null character (just like printf would do), even though there is more data in the array.
- And what about the final `printString(string4);` command? (Your output may be something different like `"0123456789AB????"`.) In this case the array of characters does not contain a null character that indicates the end of the string. This is something that you would normally want to avoid since all predefined string functions from the library such as printf, strlen, strcpy, etc. rely on the null character being present. What happens is since that there is no null character within the 12-character array the printString function just keeps printing whatever happens to be placed in memory after the string4 array. This is clearly unintended and wrong behaviour and will give different results on different computers. It may also crash your program.

Submission: Submit your file `arraysandstrings.c` that implements the last task on Oncourse. You can resubmit as often as you want until the deadline.

- 29/7, v4.0: Removed Double Dutch part (commands 1, 2, 3), fixed scoring.
- 4/8, v4.1: Clarified strncpy cannot deal with overlapping regions.
- 2/9, v4.2: Updated prompt.
- 12/9, v4.3: Added missing test case 6.

Input / output test cases

Long lines have been wrapped at 70 characters for legibility. When your program output is compared to the expected output lines will not be wrapped.

Case 01

Input:

```
X
q
```

Output:

```
Command? Unknown command 'X'
Command? Bye!
```


Case 02

Input:

```
p
a
q
```

Output:

```
Command? The current string is: ""
Command? The current array is:
[\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0,\0,\0,\0,\0,\0]
Command? Bye!
```


Case 03

Input:

```
s
hello
p
a
s
...
p
a
s
longer!
p
a
q
```

Output:

```
Command? Please enter a string? Command? The current string is:
"hello"
Command? The current array is:
[h,e,l,l,o,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0
,\0,\0,\0,\0,\0]
Command? Please enter a string? Command? The current string is: "..."
Command? The current array is:
[.,.,.,\0,o,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0,\0,\0,\0,\0]
Command? Please enter a string? Command? The current string is:
"longer!"
Command? The current array is:
[l,o,n,g,e,r,!,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0,\0,\0,\0,\0]
Command? Bye!
```


Case 04

Input:

```
s
this-is-a-single-string
p
a
s
this is a string until the newline
p
a
s

p
a
s
trailing white space
p
a
q
```

Output:

```
Command? Please enter a string? Command? The current string is:
"this-is-a-single-string"
Command? The current array is:
[t,h,i,s,-,i,s,-,a,-,s,i,n,g,l,e,-,s,t,r,i,n,g,\0,\0,\0,\0,\0,\0,\0]
Command? Please enter a string? Command? The current string is: "this
is a string until the ne"
Command? The current array is: [t,h,i,s, ,i,s, ,a, ,s,t,r,i,n,g,
,u,n,t,i,l, ,t,h,e, ,n,e,\0]
Command? Please enter a string? Command? The current string is: ""
Command? The current array is: [\0,h,i,s, ,i,s, ,a, ,s,t,r,i,n,g,
,u,n,t,i,l, ,t,h,e, ,n,e,\0]
Command? Please enter a string? Command? The current string is:
"trailing white space  "
Command? The current array is: [t,r,a,i,l,i,n,g, ,w,h,i,t,e,
,s,p,a,c,e, , ,\0,t,h,e, ,n,e,\0]
Command? Bye!
```


Case 05

Input:

```
s
whatever
o
a
o
e
o
x
p
a
q
```

Output:

```
Command? Please enter a string? Command? Find first occurrence of
which character? The first occurrence of 'a' is at index 2
Command? Find first occurrence of which character? The first
occurrence of 'e' is at index 4
Command? Find first occurrence of which character? The first
occurrence of 'x' is at index -1
Command? The current string is: "whatever"
Command? The current array is:
[w,h,a,t,e,v,e,r,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0
,\0,\0,\0,\0]
Command? Bye!
```


Case 06

Input:

```
s
x
o
a
o
e
o
x
p
a
s
xxxxxaaxxxxx
o
a
o
e
o
x
p
a
q
```

Output:

```
Command? Please enter a string? Command? Find first occurrence of
which character? The first occurrence of 'a' is at index -1
Command? Find first occurrence of which character? The first
occurrence of 'e' is at index -1
Command? Find first occurrence of which character? The first
occurrence of 'x' is at index 0
Command? The current string is: "x"
Command? The current array is:
[x,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0,\0,\0,\0,\0,\0]
Command? Please enter a string? Command? Find first occurrence of
which character? The first occurrence of 'a' is at index 5
Command? Find first occurrence of which character? The first
occurrence of 'e' is at index -1
Command? Find first occurrence of which character? The first
occurrence of 'x' is at index 0
Command? The current string is: "xxxxxaaxxxxx"
Command? The current array is:
[x,x,x,x,x,a,a,x,x,x,x,x,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0,\0]
Command? Bye!
```


Case 07

Input:

```
s
parterretrap
p
a
i
=
0
p
a
i
#
1
p
a
i
=
14
p
a
i
#
14
p
a
q
```

Output:

```
Command? Please enter a string? Command? The current string is:
"parterretrap"
Command? The current array is:
[p,a,r,t,e,r,r,e,t,r,a,p,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "=parterretrap"
Command? The current array is:
[=,p,a,r,t,e,r,r,e,t,r,a,p,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "=#parterretrap"
Command? The current array is:
[=,#,p,a,r,t,e,r,r,e,t,r,a,p,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0
,\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "=#parterretrap="
Command? The current array is:
[=,#,p,a,r,t,e,r,r,e,t,r,a,p,=,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "=#parterretrap#="
Command? The current array is:
[=,#,p,a,r,t,e,r,r,e,t,r,a,p,#,=,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0]
Command? Bye!
```


Case 08

Input:

```
p
a
i
=
0
p
a
s
01234567890123456789012345 67
p
a
i
A
11
p
a
i
B
29
p
a
q
```

Output:

```
Command? The current string is: ""
Command? The current array is:
[\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0,\0,\0,\0,\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "="
Command? The current array is:
[=,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\
0,\0,\0,\0,\0,\0,\0]
Command? Please enter a string? Command? The current string is:
"012345678901234567890123456 7"
Command? The current array is:
[0,1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,\0,\0]
Command? Insert which character? At what index? Command? The current
string is: "01234567890A12345678901234567"
Command? The current array is:
[0,1,2,3,4,5,6,7,8,9,0,A,1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,\0]
Command? Insert which character? At what index? Command? The current
string is: "012345678 90A12345678901234567B"
Command? The current array is:
[0,1,2,3,4,5,6,7,8,9,0,A,1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,B]
Command? Bye!
```


Case 09

Input:

```
s
hello there you
p
a
r
e
=
p
a
r
 l
+
p
a
r
xz
!#
p
a
r
h=++o+th=r=+you
.
p
a
q
```

Output:

```
Command? Please enter a string? Command? The current string is:
"hello there you"
Command? The current array is: [h,e,l,l,o, ,t,h,e,r,e,
,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=llo th=r= you"
Command? The current array is: [h,=,l,l,o, ,t,h,=,r,=,
,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=++o+th=r=+you"
Command? The current array is:
[h,=,+,+,o,+,t,h,=,r,=,+,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=++o+th=r=+you"
Command? The current array is:
[h,=,+,+,o,+,t,h,=,r,=,+,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "............."
Command? The current array is:
[.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Bye!
```


Case 10

Input:

```
s
hello there you
p
a
r
e
=
p
a
r
 l
+
p
a
r
xz
!#
p
a
r
h=++o+th=r=+you
.
p
a
q
```

Output:

```
Command? Please enter a string? Command? The current string is:
"hello there you"
Command? The current array is: [h,e,l,l,o, ,t,h,e,r,e,
,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=llo th=r= you"
Command? The current array is: [h,=,l,l,o, ,t,h,=,r,=,
,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=++o+th=r=+you"
Command? The current array is:
[h,=,+,+,o,+,t,h,=,r,=,+,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "h=++o+th=r=+you"
Command? The current array is:
[h,=,+,+,o,+,t,h,=,r,=,+,y,o,u,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Replace which characters? with which character? Command? The
current string is: "............."
Command? The current array is:
[.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,\0,
\0,\0]
Command? Bye!
```


Case 11

Input:

```
s
Bert and Ernie
R
4
9
p
s
123 456 789
R
3
8
p
s
0123456789
R
4
10
p
s
0123456789
R
0
4
p
s
0123456789
R
1
2
p
s
0123456789
R
0
10
p
q
```

Output:

```
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "Ernie and Bert"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "789 456 123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "2345678910"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "0123456789"
Command? Bye!
```


Case 12

Input:

```
s
Bert and Ernie
R
4
9
p
s
123 456 789
R
3
8
p
s
0123456789
R
4
10
p
s
0123456789
R
0
4
p
s
0123456789
R
1
2
p
s
0123456789
R
0
10
p
q
```

Output:

```
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "Ernie and Bert"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "789 456 123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "2345678910"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "0123456789"
Command? Bye!
```


Case 13

Input:

```
s
Bert and Ernie
R
4
9
p
s
123 456 789
R
3
8
p
s
0123456789
R
4
10
p
s
0123456789
R
0
4
p
s
0123456789
R
1
2
p
s
0123456789
R
0
10
p
q
```

Output:

```
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "Ernie and Bert"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "789 456 123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "4567890123"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "2345678910"
Command? Please enter a string? Command? Please enter index 1? Please
enter index 2? Command? The current string is: "0123456789"
Command? Bye!
```

