## Worksheet 16 - IO

**Exercise 1.**

Write a program that searches a word in the input text file and prints out the corresponding line with its line number. To search a word in a line, you can use the “strstr” function (<http://www.cplusplus.com/reference/cstring/strstr/>) in string.h. Assume the length of each line does not exceed the predefined buffer size.

Example:

“input.txt”

*Together with ECE120, which you should have already taken, this course gives an introduction to the design and programming of computing systems.*

*This course will focus on C programming, where each new C concept will be related to the fundamental concepts described in ECE120.*

*We will start by finishing our coverage of low-level concepts such as I/O, subroutines, and stacks in LC-3 assembly language, then move on to C.*

*We will cover basic programming concepts, functions, arrays, pointers, I/O, recursion, simple data structures, and concepts in object-oriented programming.*

*A bottom-up understanding of computing systems has proven more successful in helping students to understand advanced concepts in computing that follow in the ECE curriculum.*

*Again, this course requires that you take ECE120 first.*

*Concurrent enrollment is not acceptable.*

*If you have not met this requirement but think that you should still enroll, speak with the instructor.*

Input file name: input.txt

Search: 120

Found "120" in line number 1: Together with ECE120, which you should have already taken, this course gives an introduction to the design and programming of computing systems.

Found "120" in line number 2: This course will focus on C programming, where each new C concept will be related to the fundamental concepts described in ECE120.

Found "120" in line number 6: Again, this course requires that you take ECE120 first.

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define BUF 200  int main(){  FILE \*fp;  char filename[BUF], word[BUF], line[BUF];  int linenum=0;  printf("Input file name: ");  scanf("%s", filename);  printf("Search: ");  scanf("%s", word);  /\* Your code here \*/  fclose(fp);  return 0;  } |

**Exercise 2.**

Modify the above program to search and replace words. The program reads a source file and outputs a new file after replacing the given word. The program replaces the first occurrence of the word in a line.

Example:

“input.txt”

*The rose is a rose,*

*And was always a rose.*

“onput.txt”

*The lily is a rose,*

*And was always a lily.*

Enter the input file name: input.txt

Enter the output file name: output.txt

search: rose

replace: lily

|  |
| --- |
| int main(){  FILE \*fpin, \*fpout;  char srcname[BUF], desname[BUF], word[BUF], line[BUF], rep[BUF], newstr[BUF];  char \*ptr;  int len;  printf("Enter the input file name: ");  scanf("%s", srcname);  printf("Enter the output file name: ");  scanf("%s", desname);  printf("search: ");  scanf("%s", word);  printf("replace: ");  scanf("%s", rep);  /\* Your code here \*/  fclose(fpin);  fclose(fpout);  return 0;  } |