This project is submitted jointly with your partner.

Administration:

1. **Create a document that you submit to your instructor.**
   1.1. Copy the Microsoft Word © template, identified as **p01 Project Template.docx** to your Windows PC.
   1.2. Change the name of the document
      1.2.1. If you are submitting this project with partner, change the name to *CRN-lastname1- lastname2*-**p01.docx**. For example if Charles Babbage and Alan Turing are submitting a project, and they are enrolled in CRN 12345, the name of their document would be **12345-Babbage-Turing-p01.docx**. Please note that the last names of the partners are place in the name of document alphabetically.
      1.2.2. If you are submitting this project without a partner, change the name to *CRN-lastname-firstname*-**p01.docx**. For example if Alan Turing is submitting a project, and he is enrolled in CRN 12345, the name of his document would be **12345-Turing-Alan-p01.docx**.
   1.3. Change the author identification block.
      1.3.1. If you are submitting this project with partner, employ the team identification block shown below.

Team Identification Block

Author:	Mr. Charles Babbage
Student ID:	*00000000
E-Mail:	cbabbage@uco.edu
Author:	Mr. Alan Turing
Student ID:	*00000001
E-Mail:	aturing@uco.edu
Course:	CMSC 2833 – Computer Organization I
CRN:	12345
Project:	p01
Due	October 1, 2018
Account:	tt000

Be sure to change the authors to identify the partners submitting the project. Change the CRN to our current CRN. Change the date to the date given in our course administration document designating when this project is due. Change the account to the account where the project is actually stored, one of the accounts assigned to members of the partnership earlier this semester.

1.3.2. If you are submitting this project with partner, employ the team identification block shown below.

Author Identification Block
Author: Mr. Charles Babbage
Student ID: *00000000
E-Mail: cbabbage@uco.edu
Course: CMSC 2833 – Computer Organization I
CRN: 12345
Project: p01
Due October 1, 2018
Account: tt000

Be sure to change the author to identify yourself. Change the CRN to our current CRN. Change the date to the date given in our course administration document designating when this project is due. Change the account to the account assigned to you earlier this semester.

1.4. Do not change the scoring block. It is perfect just the way it is and if you change it you won't receive a perfect score.

1.5. Copy source file **p01.cpp** that you created and tested in your student account on the department computer to the following page of the submission document.

2. Submitting your project.

2.1. Create a note to your instructor (trturner@uco.edu) .

2.2. The subject should be *CRN-lastname1- lastname2***-p01** if you have a partner and *CRN-lastname-firstname-***p01** if you do not have a partner.

2.3. Attach a copy of the document you created above to the note and send the note to your instructor.

3. Creating file p01.cpp.

3.1. Create an author identification comment having the same information that is in the author identification block discussed above.

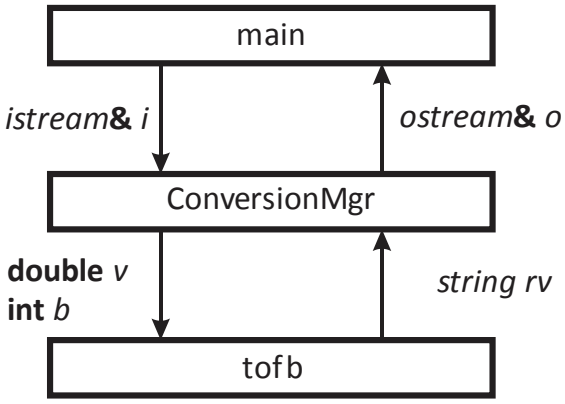
3.2. Name the file **p01.cpp**. No other name is acceptable.

3.3. Store file **p01.cpp** in the **root folder of your student account**. No other location is acceptable.

Building your solution:

1. **Command Line Arguments**
   1.1. Read and apply the discussion of command line arguments given in Lecture 1. Command Line Arguments, Programming II. (http://cs2.uco.edu/~trt/cs2613/L001.pdf)

2. **Structuring your program**



Function	Description
main	Function *main* processes command line arguments, opens the input and output file streams, invokes function *ConversionMgr* to convert values in the input file stream to corresponding values in foreign bases that are stored in the output file stream, and closes files. Function *main* catches any exceptions that are thrown and advises the user of the errors found.
ConversionMgr	Function *ConversionMgr* scans the input file for pairs of values. The first value in the pair is a rational decimal number and the second is an integer designating the foreign base. Function *ConversionMgr* terminates when the end-of-file is found or throws an exception when attempting to scan for the second value in the pair and finds **eof** instead. Function *ConversionMgr* validates that the value to convert and the foreign base fall within acceptable ranges. Function *ConversionMgr* calls function *tofb* for every valid pair and receives the converted value as a string returned from function *tofb*.
tofb	Function *tofb* splits the input value *v* into an integer portion, *ip*, and a fractional portion *fp*. The radix multiply method is employed to create a string of digits that represent the fractional portion of the converted value in the foreign base. The foreign base and the radix are identical and are specified by the input parameter *b*. The radix divide method is employed to create a string of digits that represent the integer portion in the foreign base. The integer portion, a radix point, and the fraction portion are concatenated and returned to the caller.

```
//-----------------------------------------------------------------------
//Author:      Ms. Petunia Perfect
//E-Mail:      pperfect@uco.edu
//Student ID:  *00000000
//Course:      CMSC 2833 – Computer Organization I
//CRN          21175, Spring, 2017
//Project:     p01
//Due:         February 22, 2017
//Account:     tt000
//-----------------------------------------------------------------------
```

Figure 1. Author Identification Comment to be inserted in source file p01.cpp