

**SUBMIT YOUR SOLUTIONS TO THE CANVAS BY**  
**11:59PM ON FRIDAY, JANUARY 17**  
**THERE ARE NO DOCUMENTATION REQUIREMENTS**  
**FOR THIS PROJECT**

**<VERY IMPORTANT>**

All CPE 212 projects are auto-graded in order to provide you timely feedback. It is critical that you follow the directions for the preparation and submit your projects for grading. Failing to follow directions may result in zero credit (0 points). You will submit the file `project01.cpp` to canvas by the deadline or be given zero credit(0 points).

**Project01 Goals**

The purpose of this project is to exercise, and revisit topics introduced in CPE211. We will be going through and implementing various functions in this assignment needed to accomplish our task.

**Project01 Description**

For this project, you will complete the provided partial C++ program by implementing 4 functions to perform simple image processing operations on images stored as 15x15, two dimensional arrays of integers. The image processing operations are:

- 1) Load Image
- 2) Flip Image
- 3) Transpose
- 4) Rotate Image

The rotate image functions will be given a direction of clockwise or counter clockwise, while the flip image will be given a direction of vertical or horizontal. You will implement these functions in the `project01.cpp` file. Prototypes for these functions have already been provide in the `project01.hpp` file.

The `main()` function has already been implemented for you so that you have a common way to test your code. The `main()` function will scan your input file and execute it appropriately. All program output is performed by the code in `main.cpp`. **DO NOT INCLUDE ANY OUTPUT STATEMENTS IN YOUR `project01.cpp` FILE.**

Hint: Make your output match the provided solution.

### Running the Sample Solution on **blackhawk**

*The best description of what your code should do is the Sample Solution for the project.*

Run the sample solution by typing the following at **blackhawk** terminal window command prompt

**/home/facstaff/taf0004/CPE212SP20/Project01/project01 imagefilename**

where **imagefilename** is the name of one of the provided input files (for example, **p01input1.txt**). *Your current working directory must contain the input files for this to work.*

### Unzipping Sample Input Files on **blackhawk**

Use the Firefox browser to access Angel and download **main.cpp** and the sample input files into your **Project01** directory. At terminal window prompt, use the **unzip** utility to uncompress the files.

For example: **unzip NameOfZipFileHere**

to unzip the files into your current directory.

Since this project is worth **ten points**, you have been given five input files (which utilize the three image files) to test your program.

### Running the Preview Script on **blackhawk**

Run the preview script by typing the following in a **blackhawk** terminal window command prompt

**/home/facstaff/taf0004/CPE212SP20/Project01/preview01.bash**

**This script will run both the Sample Solution AND your project01 executable program on the complete set of input files, and it compares the outputs of the two programs line by line to identify errors in your program's outputs. Make sure that the output of your program exactly matches the output of the Sample Solution.**

# Project01 Compilation Instructions

This project consists of two C++ files and one header: [main.cpp](#), [project01.cpp](#) and [project01.hpp](#). All three have been provided, along with a [makefile](#) to help you compile your program. A [makefile](#) contains the sequence of commands required to compile and assemble (link) your executable program. The provided file works on **Blackhawk**. For this assignment, you will need to use the following command line command to build your program

`make`

which will create an executable named project01 from the provided files. If your program compiles successfully, you can then type:

`./project01 NameOfInput.txt`

To execute the program assuming that the input file is located in the same directory. Example:

`./project01 p01input1.txt`

## Project 01 Specifications

```
10  enum Direction
11  {
12      D_Horizontal = -1, // causes a horizontal flip around the center line
13      D_Vertical = 1     // causes a vertical flip around the center line
14  };
15
16  enum Rotation
17  {
18      R_Clock = -1,      // causes a 90 degree rotation Clockwise
19      R_CounterClock = 1 // causes a 90 degree rotation CounterClockwise
20  };
21
22  typedef int Image[MAX_ROWS][MAX_COLS];
```

```

25 // load the image matrix from the file described.
26 // if an error is ever encountered, this should return false
27 // if the image is successfully loaded with 0 errors, this
28 // should return true.
29 bool LoadImage(const std::string & imageFile, Image image);
30
31 // This will flip the matrix according to the direction provided.
32 // a value of false is returned when an unknown Flip value is given
33 // otherwise a value of true is returned.
34 bool Flip(Image image, Direction d);
35
36 // this will rotate the matrix according to the direction provided.
37 // a value of false is returned when an unknown Rotation value is given
38 // otherwise a value of true is returned.
39 bool Rotate(Image image, Rotation r);
40
41 // this will transpose the image, which flips it across the diagonal.
42 // this diagonal goes from the top left to the bottom right.
43 void Transpose(Image image);

```

## Project01 Sample (15x15) image file

```

2 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0
1 2 1 0 0 0 0 0 0 0 0 0 0 1 0 0
0 1 6 1 0 0 0 0 0 0 0 0 1 0 0 0
0 0 1 7 1 0 0 0 0 0 0 1 0 0 0 0
0 0 0 1 7 1 0 0 0 0 1 0 0 0 0 0
0 0 0 0 1 6 1 0 1 0 0 0 0 0 0 0
0 0 0 0 0 1 5 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 5 1 0 0 0 0 0 0 0
0 0 0 0 0 1 0 1 7 1 0 0 0 0 0 0
0 0 0 0 1 0 0 0 1 7 1 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 1 9 1 0 0 0
0 0 1 0 0 0 0 0 0 0 0 1 7 1 0 0
0 1 0 0 0 0 0 0 0 0 0 0 1 6 1 0
1 0 0 0 0 0 0 0 0 0 0 0 0 1 2 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4

```

# Important Project01 Submission Instructions

You are provided the `main()` function within the `main.cpp` file. This is where all the test files will be ran to test your code. Through the program's execution, it will run various functions that you will implement within the `project01.cpp` file. **YOU WILL ONLY SUBMIT THIS PROJECT01.CPP FILE TO CANVAS.**

## THE FOLLOWING ARE VERY IMPORTANT INSTRUCTIONS:

- 1) *Do not modify the `main.cpp` file*

*Modifying the `main.cpp` file will result in zero credit (0 points) on this assignment*

- 2) *Do not modify the header file `project01.hpp`.*

*Modifying the `project01.hpp` file will result in zero credit (0 points) on this assignment*

- 3) *All of your work must be done within the `project01.cpp` file.*
- 4) *All output is done in the `main.cpp` file. Adding your own outputs to `project01.cpp` will result in zero credit (0 points).*
- 5) *Do not include the statement using `namespace std;` in your code. Any time you reference an item from the standard library, you should use `std::name`. Example, if you need to declare a string, you should declare it as `std::string myString;`*
- 6) **ONLY SUBMIT THE `project01.cpp` FILE. DO NOT SUBMIT ANY OTHER FILES WITH YOUR SUBMISSION.**