Task 1 :

Flipping the Image

Here is the code explanation:

```
#include <stdio.h>
#include <stdint.h>
typedef uint8_t BYTE;
//Here I used uint8_t to store 8 bits values or bytes


int main()
{
   FILE *src = fopen("source.tif", "rb");
   FILE *flp = fopen("flipped.tif", "wb");
//Here I opened the files

   BYTE metaData1[8];
   fread(metaData1, sizeof(BYTE), 8, src);
   fwrite(metaData1, sizeof(BYTE), 8, flp);

//Here I created a BYTE array with the size of start metadata. Then I copy pasted the metadata into the destination file.
```

```
BYTE srcRow[1192];
BYTE flpRow[1192];
for (int i = 0; i < 500; i++)
{
   fread(srcRow, sizeof(BYTE), 1192, src);
   for (int j = 1191, k = 0; j >= 0; j--, k++)
   {
      flpRow[k] = srcRow[j];
   }
   fwrite(flpRow, sizeof(BYTE), 1192, flp);
}
```

//Here I created BYTE arrays to store the values of each rows of source and flipped image. I iterated through the rows of source image to read the rows and stored the values in reversed index in the flpRow and written it to the flipped image

```
fseek(src, 0, SEEK_SET);
int bytes = 0;
BYTE b;
while (fread(&b, sizeof(BYTE), 1, src) != 0)
{
   bytes++;
}
```

//I moved the cursor back to the starting point to count total bytes.

```
    fseek(src, 0, SEEK_SET);
    int meta2Size = bytes - (8 + (1192 * 500));
    BYTE metaData2[meta2Size];
    fseek(src, 8 + (1192 * 500), SEEK_SET);
    fread(metaData2, sizeof(BYTE), meta2Size, src);
    fwrite(metaData2, sizeof(BYTE), meta2Size, flp);
//I calculated the remaining bytes left to read in the source file which is the size of end metadata. Then I copy pasted the metadata the same way of start metadata.

    fclose(src);
    fclose(flp);
    return 0;
}
```

Output:

Task 2:

Blurring the text:

```
//Here I copy pasted the meta data same way as the task 1. So no changes there.
#include <stdio.h>
#include <stdint.h>
typedef uint8_t BYTE;
int main()
{
   FILE *src = fopen("source.tif", "rb");
   FILE *blr = fopen("blurred.tif", "wb");

   BYTE metaData1[8];
   fread(metaData1, sizeof(BYTE), 8, src);
   fwrite(metaData1, sizeof(BYTE), 8, blr);

// I used 3 * 3 blurring method to blur the image
   BYTE image[500][1192];
   fread(image, sizeof(BYTE), 500 * 1192, src);
   BYTE blurred[500][1192];
//I created 2d arrays to store the values of the source image and blurred image

   for (int i = 1; i < 500; i++)
   {
      for (int j = 1; j < 1191; j++)
      {
```

```
        int sum = 0;
        for (int m = -1; m <= 1; m++)
        {
            for (int n = -1; n <= 1; n++)
            {
                sum += image[i + m][j + n];
            }
        }

        blurred[i][j] = sum / 9;
    }
}
```

//Here I iterated through the rows and columns of the image array except the values at the borders. I took the sum of the neighboring values of each values by iterating them from left to right and up to down and then calculated the average of the values and store the average values into the array for blurred image.

```
for (int i = 0; i < 1192; i++)
{
    blurred[0][i] = image[0][i];
    blurred[499][i] = image[499][i];
}
for (int j = 0; j < 500; j++)
{
    blurred[j][0] = image[j][0];
    blurred[j][1191] = image[j][1191];
```

```
    }
// I didn't change the border values, copy pasted them unchanged

    fwrite(blurred, sizeof(BYTE), 500 * 1192, blr);

    fseek(src, 0, SEEK_SET);

    int bytes = 0;
    BYTE b;

    while (fread(&b, sizeof(BYTE), 1, src) != 0)
    {
        bytes++;
    }

    fseek(src, 0, SEEK_SET);

    int meta2Size = bytes - (8 + (1192 * 500));
    BYTE metaData2[meta2Size];

    fseek(src, 8 + (1192 * 500), SEEK_SET);
    fread(metaData2, sizeof(BYTE), meta2Size, src);
    fwrite(metaData2, sizeof(BYTE), meta2Size, blr);

    fclose(src);
    fclose(blr);
```

```
    return 0;
}
```

Output: