

.TIF file format:

Initial 8 bytes are metadata.

Height and Width information can be found on file property.

The file contains trailing metadata after the picture grid.

### Task – 1: Flip the image along the horizontal axis.

#### STEPS:

- 1) Initializing the constraints.

```
const int Height = 500;  
const int Width = 1192;  
const int initial_metadata = 8;
```

- 2) Opening Files in binary read mode for input and binary write mode for output.

```
FILE* source = fopen("100dollars.tif", "rb");  
FILE* destination = fopen("200dollars.tif", "wb");
```

- 3) Copying the initial metadata from source to destination file.

```
for(int i = 0; i < initial_metadata; i++) {  
    c = fgetc(source);  
    fputc(c, destination);  
}
```

- 4) The grid information is saved in a 2D array.

```
for(int i = 0; i < Height; i++){  
    for(int j = 0; j < Width; j++){  
        grid[i][j] = fgetc(source);  
    }  
}
```

- 5) Flipping the image through horizontal axis by iterating rows from the final row to the first row and saving them in destination file.

```
    for(int i = Height - 1; i >= 0; i--){  
        for(int j = 0; j < Width; j++){  
            fputc(grid[i][j], destination);  
        }  
    }
```

- 6) Finally, the trailing metadata are copied into the destination file and both the source and destination files are closed.

```
    while(1){  
        c = fgetc(source);  
        if(c == EOF) break;  
        fputc(c, destination);  
    }  
    fclose(source);  
    fclose(destination);
```

## Task – 2:

### STEPS:

- 1) Initializing the constraints.

```
const int Height = 500;
const int Width = 1192;
const int BlurBox = 3;
const int initial_metadata = 8;
```

- 2) Opening Files in binary read mode for input and binary write mode for output.

```
FILE* source = fopen("100dollars.tif", "rb");
FILE* destination = fopen("blurdollars.tif", "wb");
```

- 3) Copying the initial metadata from source to destination file.

```
for(int i = 0; i < initial_metadata; i++) {
    c = fgetc(source);
    fputc(c, destination);
}
```

- 4) The grid information is saved in a 2D array.

```
for(int i = 0; i < Height; i++){
    for(int j = 0; j < Width; j++){
        grid[i][j] = fgetc(source);
    }
}
```

- 5) Using the method Blur Box, where we replace each grid cell with the average of its neighbours (3x3 grid) except for the border cells, we blur the image and saving them in destination file.

```

for(int i = 0; i < Height; i++) {
    for(int j = 0; j < Width; j++) {
        if(i == 0 || j == 0 || i == Height - 1 || j == Width - 1) fputc(grid[i][j], destination);
        else {
            int sum = 0;
            for(int row = i - 1; row <= i + 1; row++) {
                for(int col = j - 1; col <= j + 1; col++) sum += grid[row][col];
            }
            fputc(sum / (BlurBox * BlurBox), destination);
        }
    }
}

```

- 6) Finally, the trailing metadata are copied into the destination file and both the source and destination files are closed.

```

while(1) {
    c = fgetc(source);
    if(c == EOF) break;
    fputc(c, destination);
}
fclose(source);
fclose(destination);

```

Given Image in .tif file format:



Task – 1: Flip the image along the horizontal axis.



Task – 2: Blur the image (using Blur Box Method).

