**CS112**: Programming - 1 Assignment 3 –20th April

Cairo University, Faculty of Computers and Artificial Intelligence

## FACULTY OF COMPUTERS AND ARTIFICAL INTELLIGENCE, CAIRO UNIVERSITY

**CS112: Structured Programming Winter 2021 – 2022 Second Semester**

Assignment 3

Part 1+part 2

## Course Instructor:

Dr. Mohammed El-Ramly

### Team Members:

**20210202** Abdelrhman Hamdy Ahmed Ramadan (s7,s8) Group A

**20210398** Mostafa mahmoud basyoni Mohamed (s7,8) Group A

**20210535** Ahmed Ashraf Abd El-Hamid Khalil (s7,s8) Group A

# 1) 20210535

Filter 2: Invert Image.

1. Create a void function .
2. Create a nested loop on the photo , iterator “i” on rows and iterator “j” on column .
3. We subtract 255 from the photo to change the color from i and j .

# Filter 5: Rotate Image.

### 90 degree

1. Create a void function .
2. Create a nested loop on the photo , iterator “i” on rows and iterator “j” on column and iterator “k” equal to size of photo and k decreasing by 1.
3. We use the function (swap) to swap the rows to columns .

### 270 degree

1. Create a void function .
2. Create a nested loop on the photo , iterator “i” on rows and iterator “j” on column and iterator “k” equal to size of photo and k decreasing by 1.
3. We use the function (swap) to swap the columns to rows .

### 180 degree

1. Create a void function .
2. Create a for loop that looping twice and put the nested loop for function of 90 degree .
3. Create a nested loop on the photo , iterator “i” on rows and iterator “j” on column and iterator “k” equal to size of photo and k decreasing by 1.
4. We use the function (swap) to swap the rows to columns .

# Filter 8: Enlarge Image.

# 1- create a void function .

# 2- create a if condition (if quart = 1) that include a for loop ..when the photo begin with (0,0) and we need the get the first quart that we divide a SIZE / 2 .

# 3- (if quart = 2) we need to extract the second quart and put, enlarge it in the new image….create a for loop that the photo begin with (0,128) .

# 4- (if quart = 3) we need to extract the third quart and put, enlarge it in the new image…create a for loop that the photo begin with (128,0) .

# 5- (if quart = 4) we need to extract the forth quart and put, enlarge it in the new image…create a for loop that the photo begin with (128,128) .

# Filter b: shuffle Image.

# D:\Learning\Assignment 3 -- Programming\Gray-Scale-Image-Processor\4_22_2022 11_40_33 PM.pngWhen the user choose the new order of quarters , say that (x , y , z , a)…. x has 4 conditions, if user insert first quart of second or third or forth and so on, on the other orders y, z and a .

# 2) 20210202

Filter 1: Black and White .

1. make a function void to filter grey image to black and white :

#we make a two for loop to loop on each element of colmn and row in the array of 2D

for (int x = 0; x < SIZE; x++) { for (int y = 0; y< SIZE; y++) {

#we chick her if this pixles is greater than 127 (that means its grey (a degree of colour between black and white) ) and we make it white :

if (image[x][y] > 127)

image[x][y] = 255;

#else we make this pixles black . else

image[x][y] = 0;

}

}

}

Filter 7: Detect Image.

1. we turn the grayscale image to black and white image

2. we for loop of each element in row and each element of collmn

for (int i = 0; i < SIZE; i++) {

for (int j = 0; j < SIZE; j++) {

3. we check if the element of colmon is not equal to the next element of clomn with (the same row)

if (image[i][j] != image[i][j + 1] || image[i][j] != image[i][j - 1] || image[i][j] != image[i + 1][j]) {

4. make this pixel black .

image2[i][j] = 0;

5. if not we make it white

} else {

image2[i][j] = 255;

}

}

}

# Filter 4: Flip Image .

we make void function to flip the image

#we make a two for loop to loop on each element of colmn and row in the array of 2D

for (int x = 0 ; x <= SIZE /2 ; x++) { for (int y = 0; y != SIZE ; ++y ) {

#here we swap the first element of colmn with the last and for raw too swap(image[x][y], image[SIZE - 1 - x][y]) ;

}

}

Filter a: Mirror Image.

int choice ;

cout << " 1- mirror left " << endl << "2- mirror right" << endl << "3- mirror up" << endl << "4- mirror down" << endl ;

cin >> choice ;

if (choice == 1){

# here we mirror left so we will swap between colmns

1. we for loop of each element in row and each element of collmn.

for (int row = 0; row < SIZE ; row++)

{

for (int col = 0; col < SIZE/2 ; col++)

{

2. we swap the first element(pixel) of colmn with the same(row) with the last one and so one .

image[row][col] = image[row][255-col] ;

}

}

}

else if (choice == 2){

# we here Mirror right so we will swap with colmns

for (int row = 0; row < SIZE ; row++)

{

for (int col = 0; col < SIZE ; col++)

{

3 . we swap the last element(pixel) of colmn with the same(row) with the first and so on .

image[row][255-col] = image[row][col] ;

}

}

}

else if (choice == 3) {

# we here mirror up so we swap between rows

for (int row = 0; row < SIZE ; row++)

{

for (int col = 0; col < SIZE ; col++)

{

4. we swap the last element(pixel) of row with the same(colmn) with the first and so on .

image[255-col][row] = image [col][row] ;

}

}

}

else if (choice == 4 ){

# we here Mirror down so we swap between rows

for (int row = 0; row < SIZE ; row++)

{

for (int col = 0; col < SIZE /2 ; col++)

{

5. we here swap with the first element of row to the last element with the same (colmn)

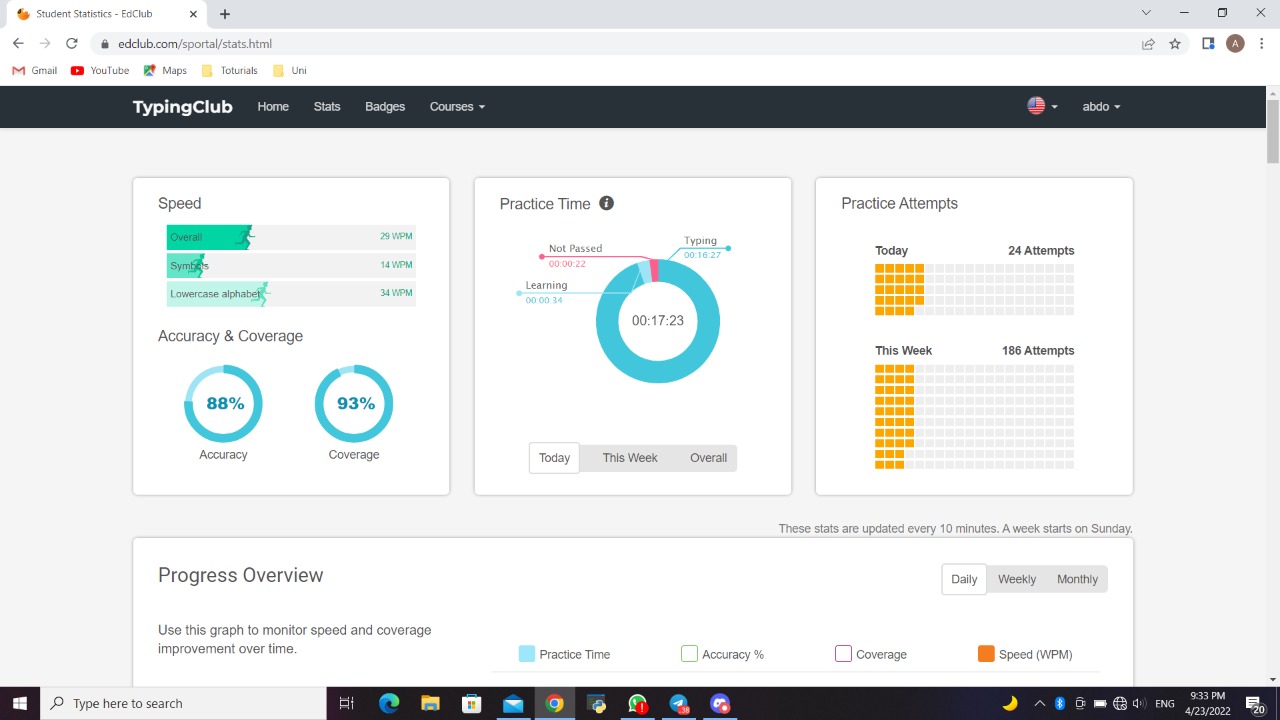
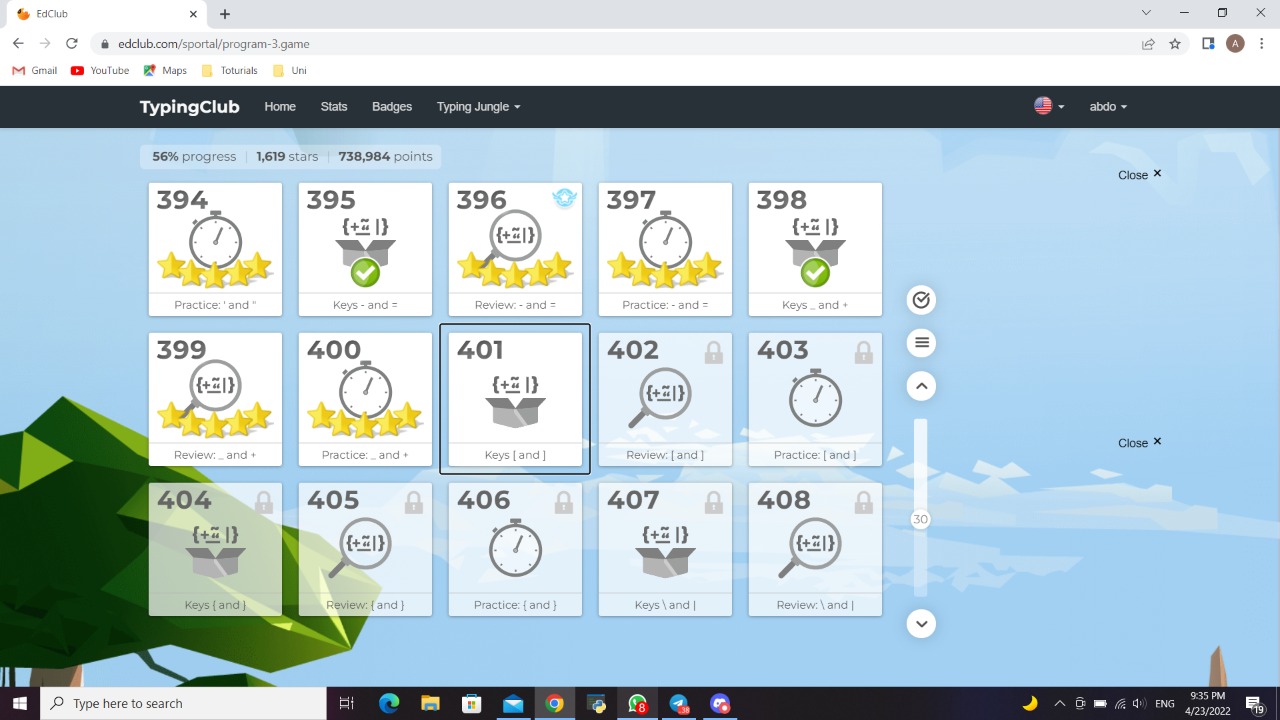
image[col][row] = image [255-col][row] ;

}

}

}

}



# 3) 20210398

Filter 3: Merge Images .

We created a new image to store the merged image in. after iteration through all of the pixels,

we used this operation: merged\_image[i][j]=image[i][j] +image2[i][j];

merged\_image is the merged image to be saved. Image is the original image. And image2 is the

second image to be merged

# Filter 6: Darken and Lighten Image .

First make a void function called Darken\_lighten\_image(). Identify variable choice as integer.

Print (“Do you want to Darken or lighten”). Input choice.

If choice = 1

For integer i = 0; i < SIZE {

For integer j = 0; j < SIZE { If image [i] [j] > 50

Subtract 50 from image [i] [j]

}

}

Else {

For i = 0; i < SIZE; {

For j = 0; j < SIZE {

If image [i] [j] < 205 Add 50 to image

}

}

}

Filter 9: Shrink Image.

void shrink()

string number;

int Wtrue;

Wtrue = true;

input "press \" 4 \" for 1/4... press \" 3 \" for 1/3...press \"2\" for 1/2: ";

print = number;

if (number == "2" || number == "3" || number == "4")

break;

else {

print "enter a valid input ! " << endl

for (int i = 0; i < SIZE; i+=2) {

for (int j = 0; j< SIZE; j+=2) {

if (number == "2")

outimage[i/2][j/2] = image1[i][j];

else if (number == "3")

outimage[i/3][j/3] = image1[i][j]

else if (number == "4")

outimage[i/4][j/4] = image1[i][j]

Filter c: Blur Image.

Delcale i j

void blur ()

int average = 0;

for (int i=0 ; i < SIZE ; i++){

for (int j=0 ; j < SIZE ; j++){

for (int k = -1; k <= 1; k ++){

for (int l = -1; l <= 1 ;l++

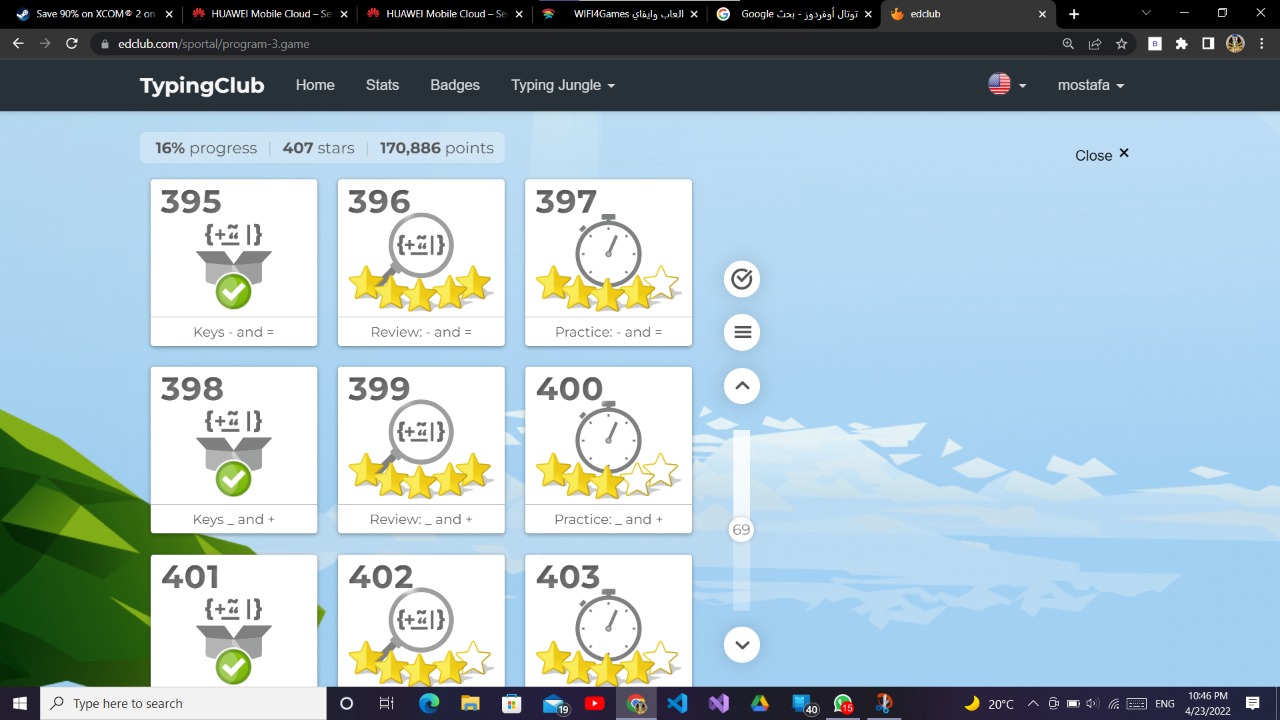
if((i + k) >= 0 && (i + k) <= 255 && (j + l) >= 0 && (j + l) <= 255){

average += image1[i + k][j + l];

printimage[i][j] = (average / 9);

making average = 0 again

average = 0;



GitHub

