Introduction to Soft Computing Assignment 1

Student: 410821305薛祖恩

Q: Write a C program to compute multiplication of two given interval matrices [A] and [B] by extending the basic multiplication of intervals.

A:

#include <stdio.h>

#include <stdlib.h>

float max(float num1, float num2)

{

return (num1 > num2) ? num1 : num2;

}

float min(float num1, float num2)

{

return (num1 > num2) ? num2 : num1;

}

int main()

{

printf("Enter the number of rows and columns in interval matrices [A] and [B]\n");

int row, col;

printf("row: ");

scanf("%d", &row);

printf("col: ");

scanf("%d", &col);

float (\*Al)[row] = malloc(sizeof(float[row][col]));

float (\*Au)[row] = malloc(sizeof(float[row][col]));

float (\*Bl)[row] = malloc(sizeof(float[row][col]));

float (\*Bu)[row] = malloc(sizeof(float[row][col]));

float (\*Cl)[row] = malloc(sizeof(float[row][col]));

float (\*Cu)[row] = malloc(sizeof(float[row][col]));

printf("\nEnter the lower (Al) bound of first interval matrix [A]:\n");

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

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

scanf("%f", &Al[i][j]);

}

}

printf("\nEnter the upper (Au) bound of first interval matrix [A]:\n");

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

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

scanf("%f", &Au[i][j]);

}

}

printf("\nEnter the lower (Bl) bound of second interval matrix [B]:\n");

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

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

scanf("%f", &Bl[i][j]);

}

}

printf("\nEnter the upper (Bu) bound of second interval matrix [B]:\n");

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

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

scanf("%f", &Bu[i][j]);

}

}

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

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

Cl[i][j] = min(min(Al[i][j] \* Bl[i][j], Al[i][j] \* Bu[i][j]), min(Au[i][j] \* Bl[i][j], Au[i][j] \* Bu[i][j]));

Cu[i][j] = max(max(Al[i][j] \* Bl[i][j], Al[i][j] \* Bu[i][j]), max(Au[i][j] \* Bl[i][j], Au[i][j] \* Bu[i][j]));

}

}

printf("\nThe multiplication of two interval matrices [A] and [B] is [C] = [A] \* [B]:\n");

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

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

printf("[%f, %f]", Cl[i][j], Cu[i][j]);

}

printf("\n");

}

return 0;

}

The input and output of the program, on page 20 and 22, the inputs are the same:

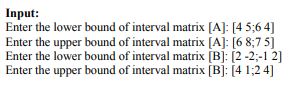
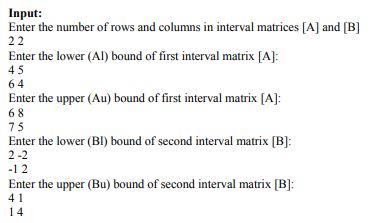


Figure1: Input of page 20 and 22

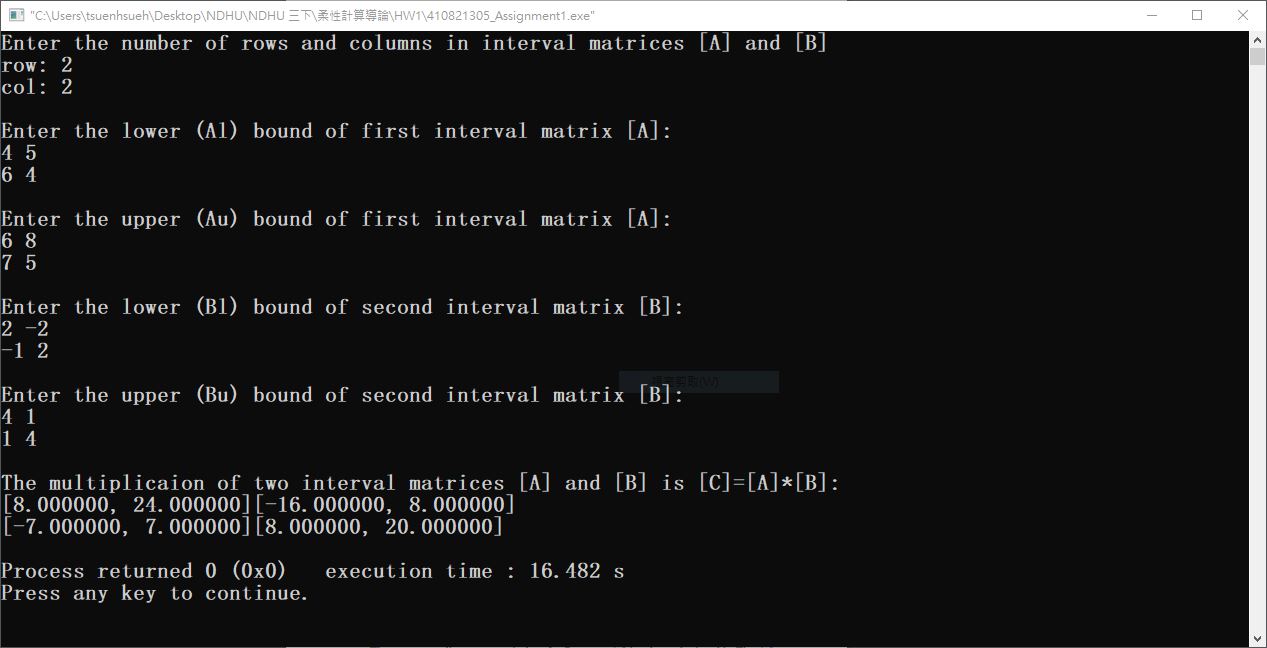


Figure2: Input and output of my program