**201533661 이승수’s algorithm homework#3 date: 2016.09.18**

**<code>**

#include <stdio.h>

struct elephant

{

int Weight;

int IQ;

}line[100];

FILE \*inF, \*outF;

int orderWeight[100] = { 0 };

int orderIQ[100] = {0};

void SortbyWeight(int count);

void SortbyIQ(int count);

void swap(int a, int b);

void main()

{

int count = 0;

inF = fopen("input.txt","r");

while (!feof(inF))

{

fscanf(inF,"%d",&line[++count].Weight);

fscanf(inF,"%d",&line[count].IQ);

}

fclose(inF);

/\*sort 2 arrays(orderWeight,orderIQ), each by weight and IQ\*/

for (int i = 1; i <= count; i++)

{

orderWeight[i] = i;

orderIQ[i] = i;

}

printf("line number sorted by Weight: ");

SortbyWeight(count);

for (int i = 1; i <= count; i++)

{

printf("%d ",orderWeight[i]);

}

printf("\nlin number sorted by IQ: ");

SortbyIQ(count);

for (int i = 1; i <= count; i++)

{

printf("%d ",orderIQ[i]);

}

/\*makes table(row(i) is ordered by weight, column(j) is ordered by IQ).\*/

int table[100][100] = { 0 };

for (int i = 1; i <= count; i++)

{

for (int j = 1; j <= count; j++)

{

if (orderWeight[i] == orderIQ[j])

table[i][j] = orderWeight[i];

}

}

for (int i = 0; i <= count; i++)

{

printf("\n");

for (int j = 0; j <= count; j++)

{

printf("%d ",table[i][j]);

}

}

outF = fopen("output.txt","w");

int num = 0;

int smartElephant[100];

int iMin = 0, jMax = count;//iMin remember last minimum weight, jMax remember last max IQ

/\*find which lines are ascending in weight and descending in IQ\*/

for (int j = jMax; j > 0;j--)

{

for (int i = (iMin + 1); i < count;i++)

{

if (table[i][j] != 0)

{

iMin = i;

jMax = j;

smartElephant[++num] = table[i][j];

}

}

}

fprintf(outF,"%d\n",num);

for (int i = 1; i <= num; i++)

{

fprintf(outF,"%d\n",smartElephant[i]);

}

fclose(outF);

}

void SortbyWeight(int count)

{

for (int i = 1; i <= count;i++)

{

int k = i;

for (int j = (i + 1); j <= count;j++)

{

if (line[orderWeight[k]].Weight > line[orderWeight[j]].Weight)

k = j;

}

if (i!=k)

swap(&orderWeight[i],&orderWeight[k]);

}

}

void SortbyIQ(int count)

{

for (int i = 1; i <= count; i++)

{

int k = i;

for (int j = (i + 1); j <= count; j++)

{

if (line[orderIQ[k]].IQ > line[orderIQ[j]].IQ)

k = j;

}

if (i != k)

swap(&orderIQ[i], &orderIQ[k]);

}

}

void swap(int \*a,int \*b)

{

int temp;

temp = \*a;

\*a = \*b;

\*b = temp;

}

