Задача. Дано сукупність відомостей заробітних плат підприємств у вигляді одновимірного масиву та у вигляді двовимірного масиву (для підприємства з декількома відділами).

Розробити класи :

BaseArray – є предком для одновимірного і двовимірного масиву;

OneDimensionArray -- представляє одновимірний масив;

TwoDimensionArray – представляє двовимірний масив;

DataBase – містить послідовність, у якій зберігаються як одновимірні, так і двовимірні масиви (тобто масив покажчиків типу BaseArray).

Необхідно для кожної відомості знайти:

1) максимальний елемент;

2) мінімальний елемент;

3) середнє значення;

4) визначити, чи є заробітна плата, що менше за вказану.

#pragma once

#include<string>

using namespace std;

class BaseArray

{

public:

virtual void input()=0;

virtual string toString() = 0;

virtual double getMaxSalary() = 0;

virtual double getMinSalary() = 0;

virtual double getAverageSalary() = 0;

bool conatinsSalaryLessThanValue(double value);

};

==========================

#include "stdafx.h"

#include "BaseArray.h"

BaseArray::BaseArray()

{

}

BaseArray::~BaseArray()

{

}

bool BaseArray::conatinsSalaryLessThanValue(double value)

{

return getMinSalary()<value;

}

======================

#pragma once

#include<string>

using namespace std;

#include "BaseArray.h"

class OneDimensionalArray :

public BaseArray

{

int \_length;

double\* data;

public:

OneDimensionalArray(int length, double firstSalary,...);

OneDimensionalArray(int length);

OneDimensionalArray();

void input();

string toString();

double getMaxSalary();

double getMinSalary();

double getAverageSalary();

~OneDimensionalArray();

};

=============================

#include "stdafx.h"

#include "OneDimensionalArray.h"

#include <iostream>

#include<string>

using namespace std;

OneDimensionalArray::OneDimensionalArray(int length, double firstSalary, ...):OneDimensionalArray(length)

{

double\* temp = &firstSalary;

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

{

data[i] = temp[i];

}

}

OneDimensionalArray::OneDimensionalArray(int length)

{

\_length = length;

data = new double[length];

}

OneDimensionalArray::OneDimensionalArray():OneDimensionalArray(15)

{

}

void OneDimensionalArray::input()

{

for (int i = 0; i < \_length; i++)

{

printf("Salary %d :",i);

cin >> data[i];

}

}

string OneDimensionalArray::toString()

{

string s = to\_string(data[0]);

for (int i = 1; i < \_length; i++)

{

s += ", " + to\_string(data[i]);

}

return s;

}

double OneDimensionalArray::getMaxSalary()

{

double max = data[0];

for (int i = 1; i < \_length; i++)

{

if (data[i]>max)

{

max = data[i];

}

}

return max;

}

double OneDimensionalArray::getMinSalary()

{

double min = data[0];

for (int i = 1; i < \_length; i++)

{

if (data[i]<min)

{

min = data[i];

}

}

return min;

}

double OneDimensionalArray::getAverageSalary()

{

double sum = 0;

for (int i = 0; i < \_length; i++)

{

sum += data[i];

}

return sum / \_length;

}

OneDimensionalArray::~OneDimensionalArray()

{

delete[] data;

}

==============================

#pragma once

#include "BaseArray.h"

#include"OneDimensionalArray.h"

class TwoDimensinalArray :

public BaseArray

{

int \_rowCount, \_columnCount;

double\*\* data;

//OneDimensionalArray\*\* rows;

public:

TwoDimensinalArray(int rowCount, int columnCount);

TwoDimensinalArray(int rowCount, int columnCount,double firstSalary,...);

TwoDimensinalArray();

void input();

string toString();

double getMaxSalary();

double getMinSalary();

double getAverageSalary();

~TwoDimensinalArray();

};

========================

#include "stdafx.h"

#include "TwoDimensinalArray.h"

#include<iostream>

using namespace std;

TwoDimensinalArray::TwoDimensinalArray(int rowCount, int columnCount)

{

\_rowCount = rowCount;

\_columnCount = columnCount;

data = new double\*[rowCount];

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

{

data[i] = new double[columnCount];

}

}

TwoDimensinalArray::TwoDimensinalArray(int rowCount, int columnCount, double firstSalary, ...):

TwoDimensinalArray(rowCount,columnCount)

{

double\* temp = &firstSalary;

int pos = 0;

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

data[i][j] = temp[pos];

pos++;

}

}

}

TwoDimensinalArray::TwoDimensinalArray():TwoDimensinalArray(15,15)

{

}

void TwoDimensinalArray::input()

{

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

printf("department %d employee %d salary:", i, j);

cin>>data[i][j];

}

}

}

string TwoDimensinalArray::toString()

{

string s = "";

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

s+="," + to\_string( data[i][j]);

}

s += "\n";

}

return s;

}

double TwoDimensinalArray::getMaxSalary()

{

double max = data[0][0];

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

if (data[i][j] > max)

max = data[i][j];

}

}

return max;

}

double TwoDimensinalArray::getMinSalary()

{

double min = data[0][0];

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

if (data[i][j] < min)

min = data[i][j];

}

}

return min;

}

double TwoDimensinalArray::getAverageSalary()

{

double sum= 0;

for (int i = 0; i < \_rowCount; i++)

{

for (int j = 0; j < \_columnCount; j++)

{

sum+=data[i][j] ;

}

}

return sum/(\_rowCount\*\_columnCount);

}

TwoDimensinalArray::~TwoDimensinalArray()

{

for (int i = 0; i < \_rowCount; i++)

{

delete[] data[i];

}

delete[] data;

}

=============================

#pragma once

#include"BaseArray.h"

class DataBase

{

int \_length;

BaseArray\*\* arrays;

public:

DataBase(int length);

DataBase();

void input();

void printMaxValues();

void printMinValues();

void printAverageValues();

~DataBase();

};

===============================

#include "stdafx.h"

#include "DataBase.h"

#include "OneDimensionalArray.h"

#include "TwoDimensinalArray.h"

#include <iostream>

using namespace std;

DataBase::DataBase(int length)

{

\_length = length;

arrays = new BaseArray\*[length];

}

DataBase::DataBase():DataBase(15)

{

}

void DataBase::input()

{

for (int i = 0; i < \_length; i++)

{

int c;

printf("1. One dim. \n2.Two dim.\n Your choise:");

cin >> c;

switch (c)

{

case 1:

int count;

printf("count of employees: ");

cin >> count;

arrays[i] = new OneDimensionalArray(count);

arrays[i]->input();

break;

case 2:

int rowCount, collumnCount;

printf("departments count : ");

cin >> rowCount;

printf("count of employees: ");

cin >> collumnCount;

arrays[i] = new TwoDimensinalArray(rowCount,collumnCount);

arrays[i]->input();

break;

}

}

}

void DataBase::printMaxValues()

{

for (int i = 0; i < \_length; i++)

{

printf("Company: %d , max salary: %f",i,arrays[i]->getMaxSalary());

}

}

void DataBase::printMinValues()

{

for (int i = 0; i < \_length; i++)

{

printf("Company: %d , min salary: %f", i, arrays[i]->getMinSalary());

}

}

void DataBase::printAverageValues()

{

for (int i = 0; i < \_length; i++)

{

printf("Company: %d , average salary: %f", i, arrays[i]->getAverageSalary());

}

}

DataBase::~DataBase()

{

for (int i = 0; i < \_length; i++)

{

delete arrays[i];

}

delete[] arrays;

}

====================================

// ConsoleApplication73.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include"DataBase.h"

#include<iostream>

using namespace std;

int main()

{

printf("Company count:");

int companyCount;

cin >> companyCount;

DataBase\* dataBase = new DataBase(companyCount);

dataBase->input();

cout << "-------------- max -----------" << endl;

dataBase->printMaxValues();

cout << "-------------- min -----------" << endl;

dataBase->printMinValues();

cout << "-------------- average -----------" << endl;

dataBase->printAverageValues();

system("pause");

return 0;

}