**Problem 1)**

Find number of Zeros in matrix (two-dimensional array)- use as an example:

You can input your numbers in your program for example: Array = {{10, 0, 2},{7, 8, 3},{0,0,1}};

#include "stdafx.h"

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

const int rows = 3;

const int columns = 3;

int i, j, count = 0, a[rows][columns] = {4,2,5,0,9,8,0,7,0};

int sum = rows + columns;

//cout << "Please enter " << sum << " numbers, separated by spaces, to build the array: ";

for (i = 0; i < rows; i++)

for (j = 0; j < columns; j++)

{

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

{

cout << a[i][j] << " ";

}

else count++;

}//for

cout << endl << count << " Zeros detected." << endl;

system("Pause");

return 0;

}

**Problem 2)**

Extract diagonal elements of a square matrix and put into a column array.

Example: Start with matrix of size 4 by 4 below that can be hard coded into your program:

#include "stdafx.h"

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

const int rows = 4;

const int columns = 4;

int i, j, count = 0, a[rows][columns] = {10,435,2350,340,2320,20,7247,88,99,5353,30,33,40,41,42,40};

int b[columns];

int sum = rows + columns;

cout << "Diagonal Array: \n" << a[0][0] << endl;

for (i = 0; i < rows; i++)

{

for (j = 0; j < columns; j++)

{

if (i = j)

{

b[i] = a[i][j];

cout << a[i][j] << endl;

}//if

}//for

}//for

//cout << endl << count << " Zeros detected and removed." << endl;

system("Pause");

return 0;

}

**Program 3)**

Using pointers find SUM of elements of an array.

You can input your array anyway you want. Example: array={2.1,4.8,9.7,0.2, 0.85}

**NOTE:** Use pointers to point to the array instead of using array index.

#include "stdafx.h"

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

const int n = 3;

int i;

double sum = 0, a[n];

cout << "Please enter " << n << " numbers, separated by spaces, to build the array: ";

for (i = 0; i < n; i++)

{

cin >> a[i];

}//for

for (i = 0; i < n; i++)

{

sum = sum + \*(a + i);

}

cout << "\nThe sum of the array is: " << sum << endl;

system("Pause");

return 0;

}

**Program 4)**

Modify the main() function in the following program to add a Bathroom of size (10.2, 10.5) and existing Dining room is now modified to be 10.6, 8.0 instead of 14,10.5

Add comments where you are adding or modifying the code

Your program should display Total area of the home after the modifications.

#include <iostream>

#include <iomanip>

using namespace std;

class RoomType

{

private:  
double length; //Declare length as a double variable double width; //Declare width as a double variable

public:  
RoomType(double = 0.0, double = 0.0); //The constructor's declaration void showRoomValues();  
void setNewRoomValues(double, double);  
double calculateRoomArea();

};

//Class implementation section  
RoomType::RoomType(double l, double w) //This is a constructor {

length = l;

width = w; }

void RoomType::showRoomValues() //Accessor {

cout << " Length = " << length

<< "\n width = " << width << endl;

}  
void RoomType::setNewRoomValues(double l, double w) //Mutator {

length = l;

width = w; }

double RoomType::calculateRoomArea() //Performs the area calculation {

return (length \* width);

}

int main() {

//Declares variable of type of RoomType with length x width dimensions

RoomType Hall(12.40, 3.5);  
RoomType kitchen(14, 14);  
RoomType LivingRoom(12.4, 20);  
RoomType DiningRoom(14, 10.5);  
double sum = 0; //Sum double variable to stores the area of each room after calculation

cout << "---------Hall----------------\n";  
cout << "The area of the hall is: " << hall.calculateRoomArea(); cout << endl;  
sum = sum + hall.calculateRoomArea(); //Adds hall area into sum

cout << "\n---------Kitchen-------------\n";  
cout << "The area of the kitchen is: " << kitchen.calculateRoomArea(); cout << endl;  
sum = sum + kitchen.calculateRoomArea(); //Adds kitchen area into sum

cout << "\n---------Living Room---------\n";  
cout << "The area of the living room is: " << LivingRoom.calculateRoomArea(); cout << endl;  
sum = sum + LivingRoom.calculateRoomArea(); //Adds living room area into sum

cout << "\n---------Dining Room---------\n";  
cout << "The area of the dining room is: " << DiningRoom.calculateRoomArea(); cout << endl;  
sum = sum + DiningRoom.calculateRoomArea(); //Adds dining room area into sum

cout << "\n---------All 4 Rooms---------\n";  
cout << "The total area of all 4 rooms combined is: "

<< sum; //Outputs value of each room's area that has been stored into sum

cout << endl << endl;

system("pause");

return 0;

}

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

// ConsoleApplication4.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <iostream>

#include <iomanip>

using namespace std;

class RoomType

{

private:

double width; //Added by Alex Miles on 12/1/18

double length; //Declare length as a double variable double width; //Declare width as a double variable

public:

RoomType(double = 0.0, double = 0.0); //The constructor's declaration void showRoomValues();

void setNewRoomValues(double, double);

double calculateRoomArea();

void showRoomValues();

};

//Class implementation section

RoomType::RoomType(double l, double w) //This is a constructor

{

length = l;

width = w;

}//RoomType

void RoomType::showRoomValues() //Accessor

{

cout << " Length = " << length

<< "\n width = " << width << endl;

}//Accessor

void RoomType::setNewRoomValues(double l, double w) //Mutator

{

length = l;

width = w;

}//mutator

double RoomType::calculateRoomArea() //Performs the area calculation

{

return (length \* width);

}//area calculation

int main() {

//Declares variable of type of RoomType with length x width dimensions

RoomType Hall(12.40, 3.5);

RoomType Kitchen(14, 14);

RoomType LivingRoom(12.4, 20);

RoomType DiningRoom(10.6, 8.0); //Modified from 14, 10.5 by Alex Miles on 12/1/18

RoomType BathRoom(10.2, 10.5); //Room added by Alex Miles on 12/1/18

double sum = 0; //Sum double variable to stores the area of each room after calculation

cout << "---------Hall----------------\n";

cout << "The area of the hall is: " << Hall.calculateRoomArea(); cout << endl;

sum = sum + Hall.calculateRoomArea(); //Adds hall area into sum

cout << "\n---------Kitchen-------------\n";

cout << "The area of the kitchen is: " << Kitchen.calculateRoomArea(); cout << endl;

sum = sum + Kitchen.calculateRoomArea(); //Adds kitchen area into sum

cout << "\n---------Living Room---------\n";

cout << "The area of the living room is: " << LivingRoom.calculateRoomArea(); cout << endl;

sum = sum + LivingRoom.calculateRoomArea(); //Adds living room area into sum

cout << "\n---------Dining Room---------\n";

cout << "The area of the dining room is: " << DiningRoom.calculateRoomArea(); cout << endl;

sum = sum + DiningRoom.calculateRoomArea(); //Adds dining room area into sum

cout << "\n---------Bath Room---------\n";

cout << "The area of the bath room is: " << BathRoom.calculateRoomArea(); cout << endl;

sum = sum + BathRoom.calculateRoomArea(); //Adds bath room area into sum

cout << "\n---------All 5 Rooms---------\n"; //AM 5 rooms now!

cout << "The total area of all 5 rooms combined is: "

<< sum; //Outputs value of each room's area that has been stored into sum

cout << endl << endl;

system("pause");

return 0;

}