//**04-07** f overload determines whether a round pizza or a rectangular pizza is the best buy.

#include <iostream>

#include <conio.h>

using namespace std;

//compiler overload resolving rules:

//**1.** Exact match (number and types of arguments, no automatic type conversion)

//**2.** Match using automatic type conversion (ex: int into double)

// f (int, double), f (double, int), call f(2,5) confusing-error, adding f(int,int) ,

// call (2,5) no error

double unitPrice(int diameter, double price); //round pizza, two parameters

//Returns the price per square inch of a round pizza.

//The formal parameter named diameter is the diameter of the pizza

//in inches. The formal parameter named price is the price of the pizza.

double unitPrice(int length, int width, double price); //rectangular pizza, three parameters

//Returns the price per square inch of a rectangular pizza

//with dimensions length by width inches.

//The formal parameter price is the price of the pizza.

int main( )

{

int diameter, length, width;

double priceRound, unitPriceRound, priceRectangular, unitPriceRectangular;

cout << "Welcome to the Pizza Consumers Union.\n";

cout << "Enter the diameter in inches"

<< " of a round pizza: ";

cin >> diameter;

cout << "Enter the price of a round pizza: $";

cin >> priceRound;

cout << "Enter length and width in inches\n"

<< "of a rectangular pizza: ";

cin >> length >> width;

cout << "Enter the price of a rectangular pizza: $";

cin >> priceRectangular;

unitPriceRectangular = unitPrice(length, width, priceRectangular);

unitPriceRound = unitPrice(diameter, priceRound);

cout.setf(ios::fixed);

cout.setf(ios::showpoint);

cout.precision(2);

cout << endl

<< "Round pizza: Diameter = "

<< diameter << " inches\n"

<< "Price = $" << priceRound

<< " Per square inch = $" << unitPriceRound

<< endl

<< "Rectangular pizza: Length = "

<< length << " inches\n"

<< "Rectangular pizza: Width = "

<< width << " inches\n"

<< "Price = $" << priceRectangular

<< " Per square inch = $" << unitPriceRectangular

<< endl;

if (unitPriceRound < unitPriceRectangular)

cout << "The round one is the better buy.\n";

else

cout << "The rectangular one is the better buy.\n";

cout << "Buon Appetito!\n";

system("pause");

return 0;

}

double unitPrice(int diameter, double price)

{

const double PI = 3.14159;

double radius, area;

radius = diameter/double(2);

area = PI \* radius \* radius;

return (price/area);

}

double unitPrice(int length, int width, double price)

{

double area = length \* width;

return (price/area);

}