SHARJEEL MEMON (BCS-2E)

24K-0555

FAST UNIVERSITY

LECTURER:MUHAMMAD MONIS

**OBJECT ORIENTED PROGRAMMING LAB-1**

**QUESTION NO.1**

**Q1: Write a program that finds the second highest number in a float type array of 20 elements using pointer.**

**Solution:**

#include <iostream>

#include <iomanip>

using namespace std;

void AssignData(double \*arr, int S, int F, char\* argv[], int argc) {

if (F == 0) {

// Command-line Argument Mode

for (int i = 1; i < argc; i++) {

\*(arr + i - 1) = atof(argv[i]);

}

} else {

// Interactive Input Mode

cout << "Enter 20 Elements:" << endl;

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

cout << "Element " << i + 1 << ": ";

cin >> \*(arr + i);

}

}

}

void BubbleSort(double \*arr, int S) {

for (int i = 0; i < S - 1; i++) {

for (int j = 0; j < S - i - 1; j++) {

if (\*(arr + j) > \*(arr + j + 1)) {

double temp = \*(arr + j);

\*(arr + j) = \*(arr + j + 1);

\*(arr + j + 1) = temp;

}

}

}

}

double SecondLargest(double \*arr, int S) {

BubbleSort(arr, S);

return \*(arr + S - 2);

}

int main(int argc, char \*argv[]) {

int size = 20;

int flag = (argc < 2) ? 1 : 0;

if (argc > 2 && argc < 21) {

cout << "The amount of elements should be exactly 20." << endl;

return 1;

}

double num[20];

double \*ptr = num;

AssignData(ptr, size, flag, argv, argc);

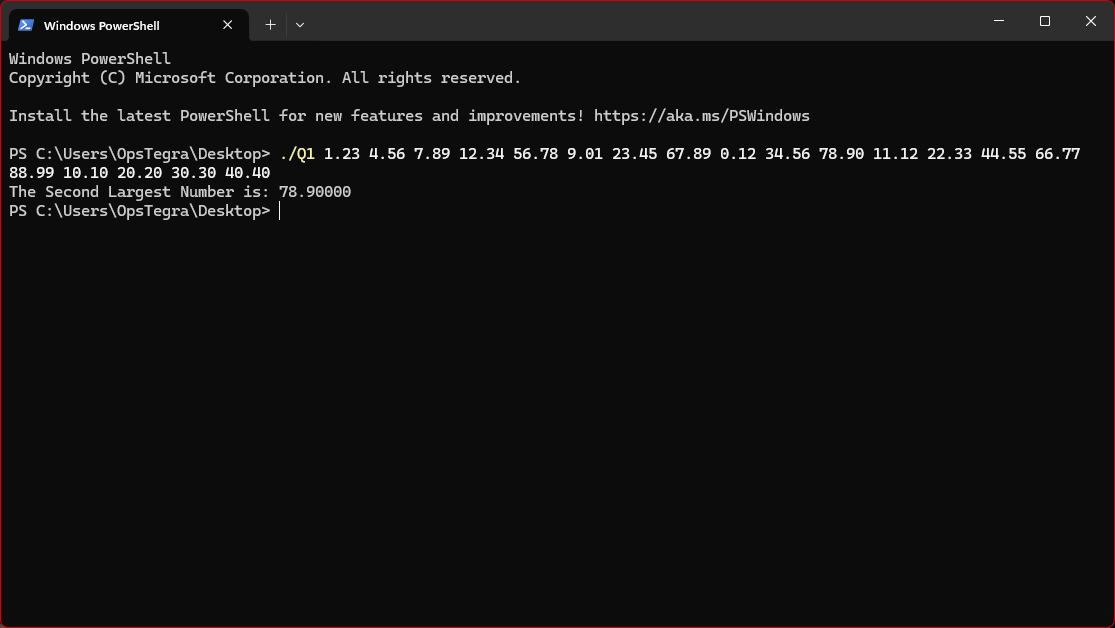
//Output

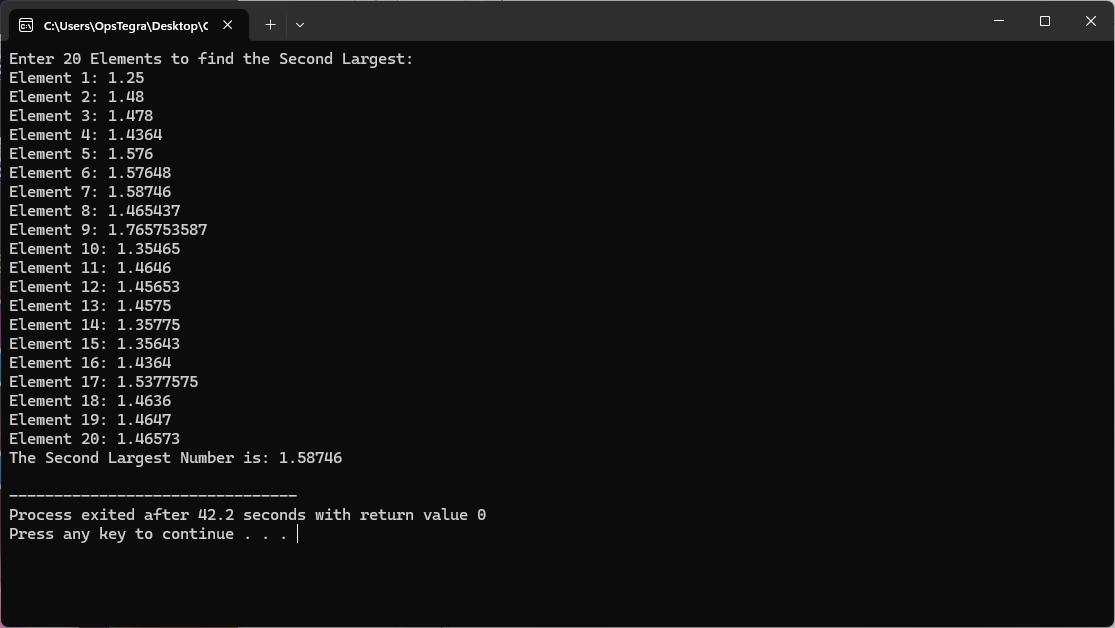
cout << "The Second Largest Number is: " << SecondLargest(ptr, size) << endl;

return 0;

}

**Result:**





**QUESTION NO.2**

**Q2: Write a program that calculates the sum of all the elements in array using pointers.**

**Solution:**

#include <iostream>

#include <iomanip>

using namespace std;

void AssignData(double \*arr, int S, int F, char\* argv[], int argc) {

if (F == 0) {

// Command-line Argument Mode

for (int i = 1; i < argc; i++) {

\*(arr + i - 1) = atof(argv[i]);

}

} else {

// Interactive Input Mode

cout << "Enter " << S << " Elements:" << endl;

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

cout << "Element " << i + 1 << ": ";

cin >> \*(arr + i);

}

}

}

double CalculateSum(double \*arr, int S) {

double sum = 0;

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

sum += \*(arr + i);

}

return sum;

}

int main(int argc, char \*argv[]) {

int size = (argc > 1) ? argc - 1 : 0;

int flag = (argc < 2) ? 1 : 0;

if (size == 0) {

cout << "Enter the number of elements: ";

cin >> size;

if (size <= 0) {

cout << "Invalid size!" << endl;

return 1;

}

}

double arr[size];

AssignData(arr, size, flag, argv, argc);

//Output

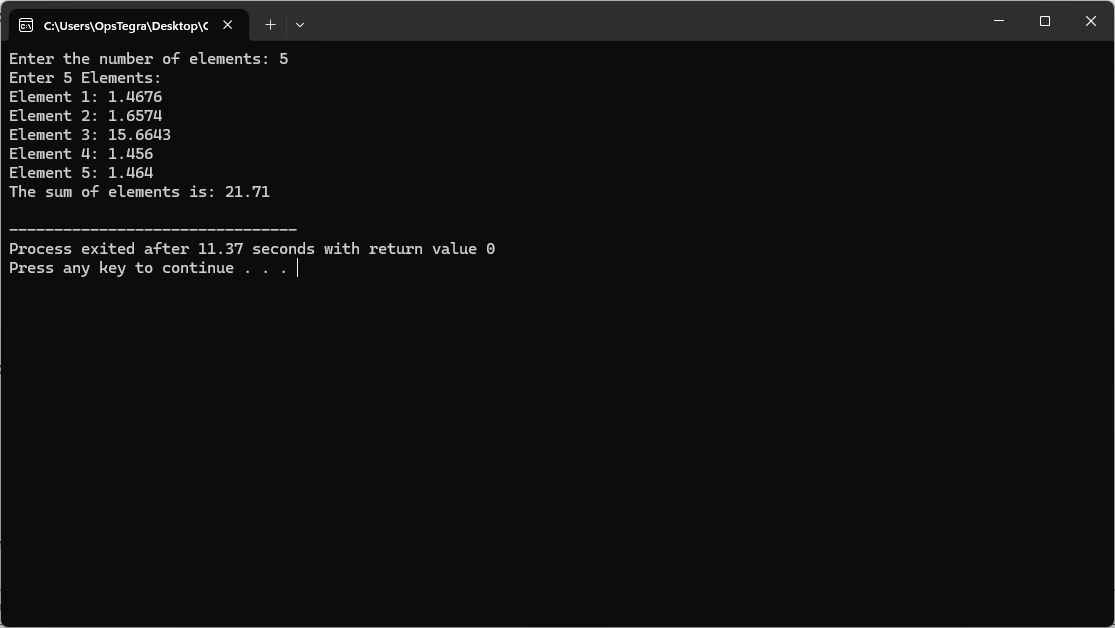
cout << fixed << setprecision(2);

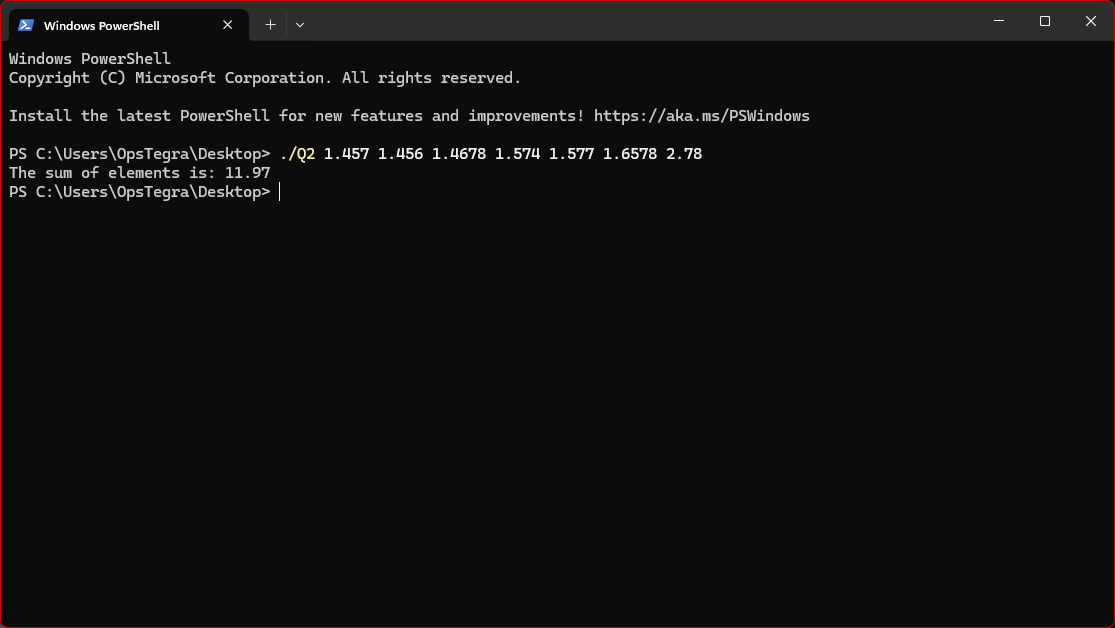
cout << "The sum of elements is: " << CalculateSum(arr, size) << endl;

return 0;

}

**Result:**





**QUESTION NO.3**

**Q3: Write a program in C++ to calculate and print the Electricity bill of a given customer. The customer id, name and unit consumed by the user should be taken from the keyboard and display the total amount to pay to the customer. The charges are as follow:**

**If bill exceeds Rs. 18000 then a surcharge of 15% will be charged on top of the bill.**

**Test Input:**

**1001 //Customer ID**

**James //Customer Name**

**800 //Units Consumed**

**Expected Output:**

**Customer ID :1001**

**Customer Name: James**

**Units Consumed :800**

**Amount Charges @Rs. 35.90 per unit: 28720**

**Surcharge Amount: 4308**

**Net Amount Paid by the Customer: 33028.00**

**Solution:**

#include <iostream>

#include <string>

#include <iomanip>

using namespace std;

float unitcalculator(int N, float &rate) {

if (N < 199) {

rate = 16.20;

return rate \* N;

}

if (N >= 200 && N < 300) {

rate = 20.10;

return rate \* N;

}

if (N >= 300 && N < 500) {

rate = 27.10;

return rate \* N;

}

if (N >= 500) {

rate = 35.90;

return rate \* N;

}

return 0;

}

int main(int argc, char \*argv[]) {

double unitsconsumed = 0;

int cusid = 0;

string cusname = "";

// For command-line argument Mode

if (argc > 1) {

if (argc < 3) {

cout << "Error: Incomplete Input!" << endl;

return 0;

}

cusid = atoi(argv[1]);

unitsconsumed = atof(argv[argc - 1]);

if (cusid <= 0 || unitsconsumed < 0) {

cout << "Error: Invalid Input! Customer ID and Units Consumed must be positive values." << endl;

return 1;

}

// Used to ensure if customers put 2 or more words for their name....

for (int i = 2; i < argc - 1; i++) {

cusname += argv[i];

if(i!=argc-2) {

cusname +=" ";

}

}

}

// For Interactive Input Mode

else {

cout << "\n--- Enter Customer Details ---" << endl;

cout << "Customer ID: ";

cin >> cusid;

if (cusid <= 0) {

cout << "Error: Invalid Input! Customer ID must be a positive number." << endl;

return 1;

}

// Used to ignore newline buffer

cin.ignore();

cout << "Customer Name: ";

getline(cin, cusname);

cout << "Units Consumed: ";

cin >> unitsconsumed;

if (unitsconsumed < 0) {

cout << "Error: Invalid Input! Units Consumed must be a non-negative value." << endl;

return 1;

}

}

//// Calculating Amount Charges and Surplus using unitcalculator function

float ratePerUnit = 0;

float AmountCharges = unitcalculator(unitsconsumed, ratePerUnit);

float surplus = (AmountCharges > 18000) ? AmountCharges \* 0.15 : 0;

float netamount = AmountCharges + surplus;

// Output

cout << "\n--- Bill Summary ---" << endl;

cout << fixed << setprecision(2);

cout << "Customer ID : " << cusid << endl;

cout << "Customer Name : " << cusname << endl;

cout << "Units Consumed : " << unitsconsumed << endl;

cout << "Amount Charges : $" << AmountCharges << " @ " << ratePerUnit << " per unit" << endl;

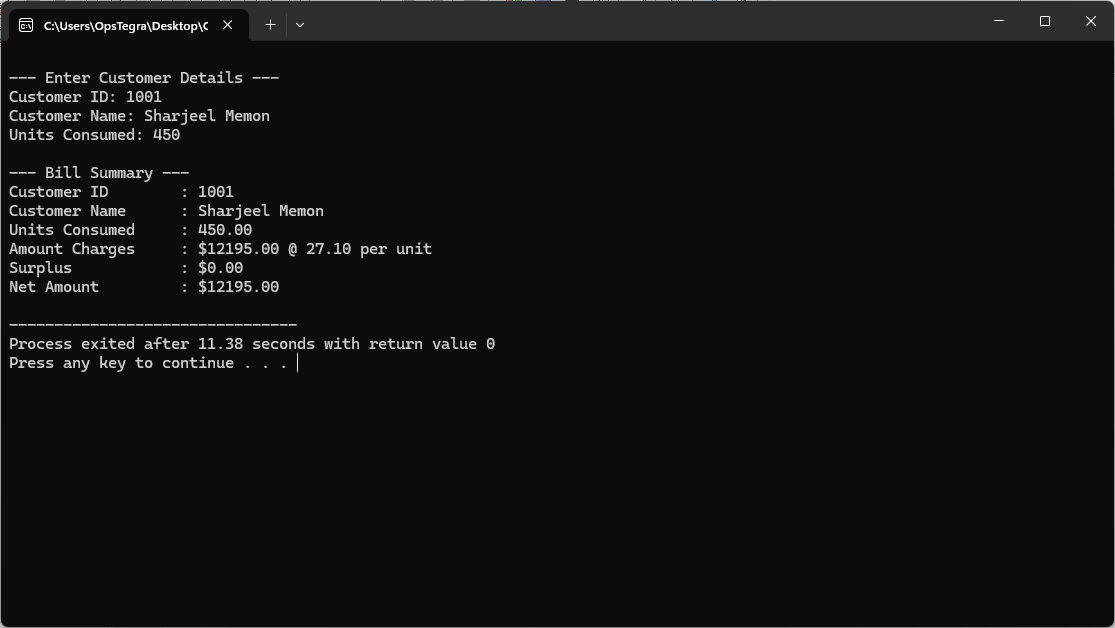
cout << "Surplus : $" << surplus << endl;

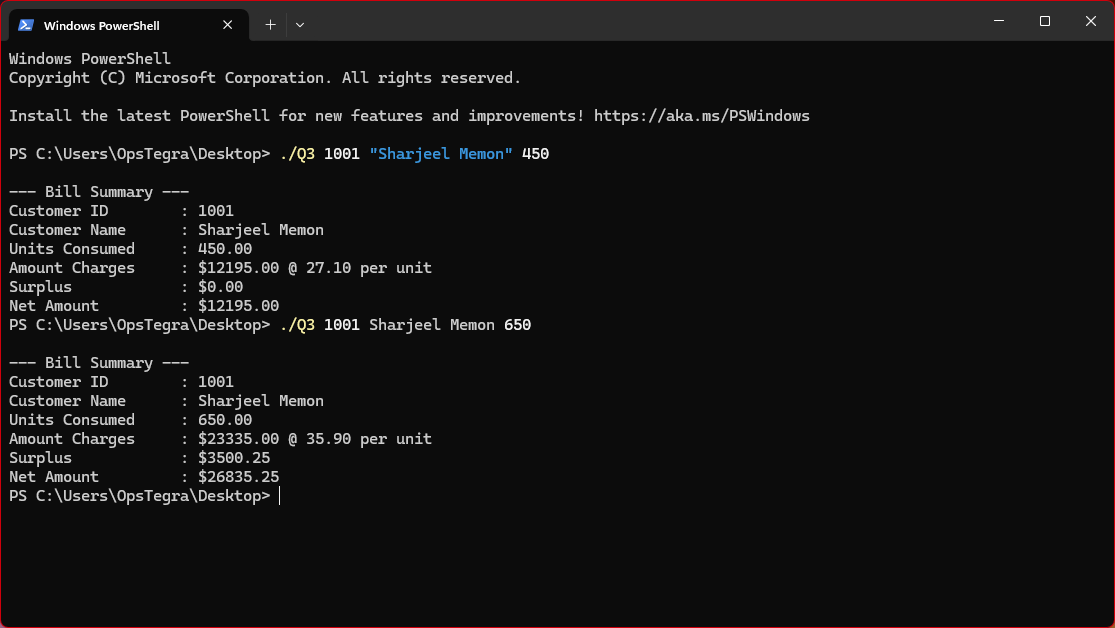
cout << "Net Amount : $" << netamount << endl;

return 0;

}

**Result:**





**QUESTION NO.4**

**Q4:Write a program that prompts the user to enter the weight of a person in kilograms and outputs the equivalent weight in pounds. Output both the weights rounded to two decimal places. (Note that 1**

**kilogram = 2.2 pounds.) Format your output with two decimal places.**

**Solution:**

#include <iostream>

#include <cstdio>

using namespace std;

int main(int argc, char \*argv[]) {

double weightinkg = 0;

//for Command-line argument mode

if (argc >= 2) {

weightinkg = stod(argv[1]);

}

//for Interactive input mode

else {

cout << "Enter your Weight in Kg: ";

cin >> weightinkg;

}

//Validating weight

if (weightinkg <= 0) {

printf("Please input a valid positive weight!\n");

return 1;

}

double ConversionFactor = 2.20462;

double weightinpound = weightinkg \* ConversionFactor;

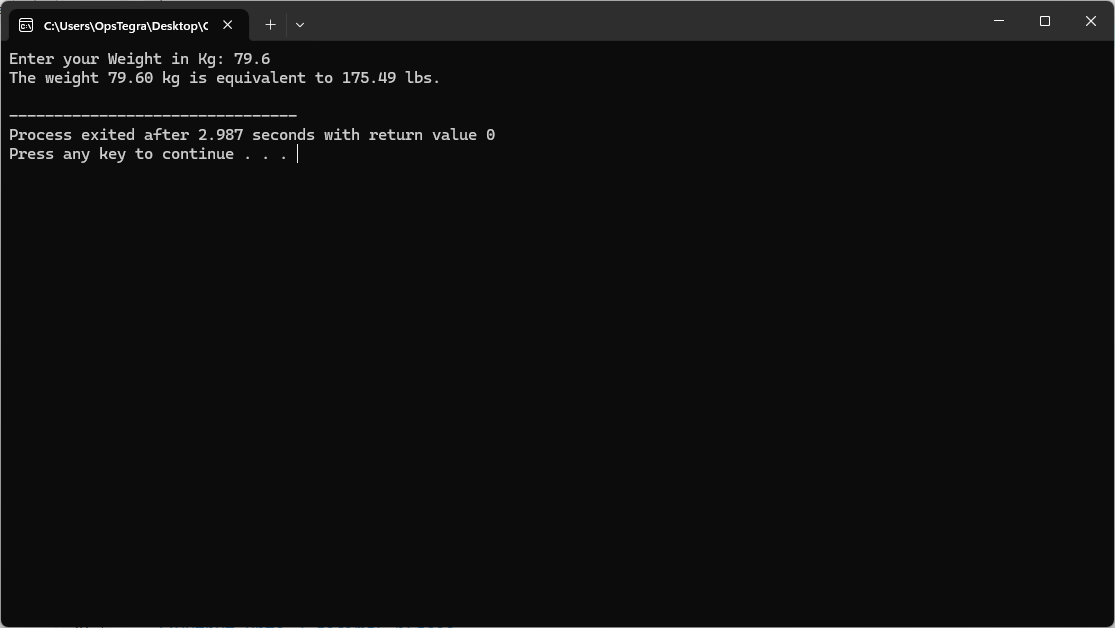
//Output upto 2 decimal places

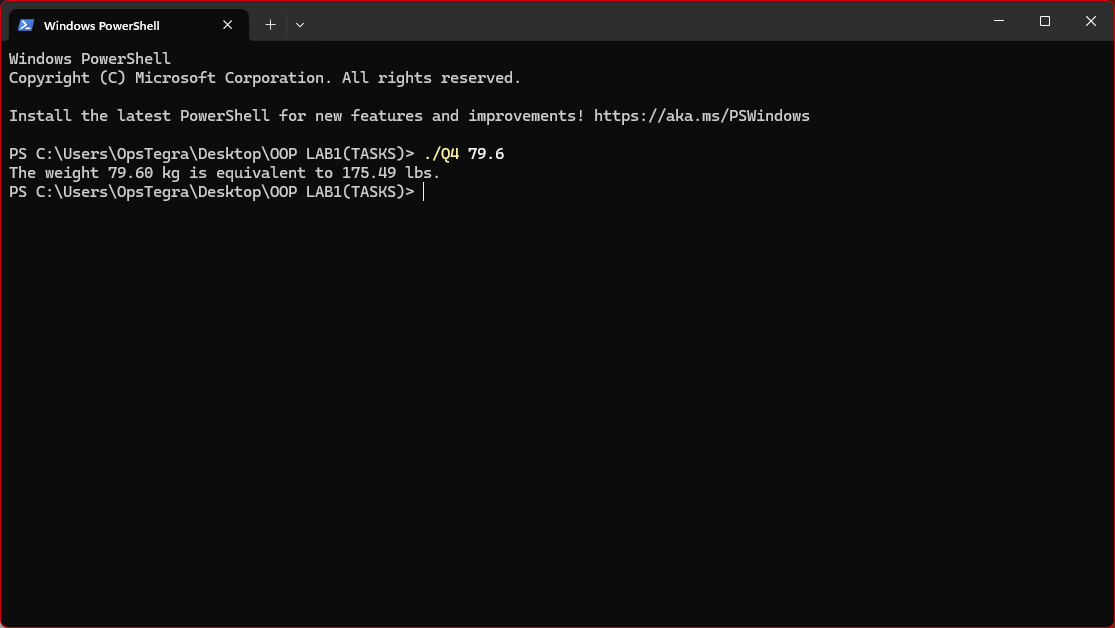
printf("The weight %.2f kg is equivalent to %.2f lbs.\n", weightinkg, weightinpound);

return 0;

}

**Result:**





**QUESTION NO.5**

**Q5:A movie in a local theater is in great demand. To help a local charity, the theater owner has decided to**

**donate to the charity a portion of the gross amount generated from the movie. This example designs and**

**implements a program that prompts the user to input the movie name, adult ticket price, child ticket price,**

**number of adult tickets sold, number of child tickets sold, and percentage of the gross amount to be**

**donated to the charity. The output of the program is as follows.**

**-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\***

**Movie Name: .................................... Journey to Mars**

**Number of Tickets Sold: .......................... 2650**

**Gross Amount: ......................................... $ 9150.00**

**Percentage of Gross Amount Donated: 10.00%**

**Amount Donated: ................................... $ 915.00**

**Net Sale: ................................................... $ 8235.00**

**Note that the strings, such as "Movie Name:" , in the first column are left-justified, the numbers in the**

**right column are right-justified, and the decimal numbers are output with two decimal places.**

**Input: The input to the program consists of the movie name, adult ticket price, child ticket price, number**

**of adult tickets sold, number of child tickets sold, and percentage of the gross amount to be donated to the**

**charity.**

**Output: The output is as shown above.**

**Solution:**

#include <iostream>

#include <iomanip>

using namespace std;

int main(int argc, char\* argv[]) {

// Variables

string movieName;

double adultTicketPrice, childTicketPrice;

int numAdultTicketsSold, numChildTicketsSold;

double donationPercentage;

if (argc == 7) {

// Command-line argument mode

movieName = argv[1];

adultTicketPrice = atof(argv[2]);

childTicketPrice = atof(argv[3]);

numAdultTicketsSold = atoi(argv[4]);

numChildTicketsSold = atoi(argv[5]);

donationPercentage = atof(argv[6]);

} else if (argc > 7) {

movieName = argv[1];

for (int i = 2; i < argc - 5; ++i) {

movieName += " " + string(argv[i]);

}

adultTicketPrice = atof(argv[argc - 5]);

childTicketPrice = atof(argv[argc - 4]);

numAdultTicketsSold = atoi(argv[argc - 3]);

numChildTicketsSold = atoi(argv[argc - 2]);

donationPercentage = atof(argv[argc - 1]);

} else {

// Interactive input mode

cout << "Enter Movie Name: ";

getline(cin, movieName);

cout << "Enter Adult Ticket Price: ";

cin >> adultTicketPrice;

cout << "Enter Child Ticket Price: ";

cin >> childTicketPrice;

cout << "Enter Number of Adult Tickets Sold: ";

cin >> numAdultTicketsSold;

cout << "Enter Number of Child Tickets Sold: ";

cin >> numChildTicketsSold;

cout << "Enter Percentage of Gross Amount to Donate: ";

cin >> donationPercentage;

}

// Calculations

int totalTicketsSold = numAdultTicketsSold + numChildTicketsSold;

double grossAmount = (adultTicketPrice \* numAdultTicketsSold) + (childTicketPrice \* numChildTicketsSold);

double amountDonated = (grossAmount \* donationPercentage) / 100;

double netSale = grossAmount - amountDonated;

// Output

cout <<endl<< "-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*" << endl;

cout << left << setw(30) << "Movie Name:" << setw(30) << movieName << endl;

cout << left << setw(30) << "Number of Tickets Sold:" << setw(20) << totalTicketsSold << endl;

cout << left << setw(30) << "Gross Amount:" << "$" << right << setw(15) << fixed << setprecision(2) << grossAmount << endl;

cout << left << setw(30) << "Percentage of Gross Amount Donated:" << right << setw(15) << fixed << setprecision(2) << donationPercentage << "%" << endl;

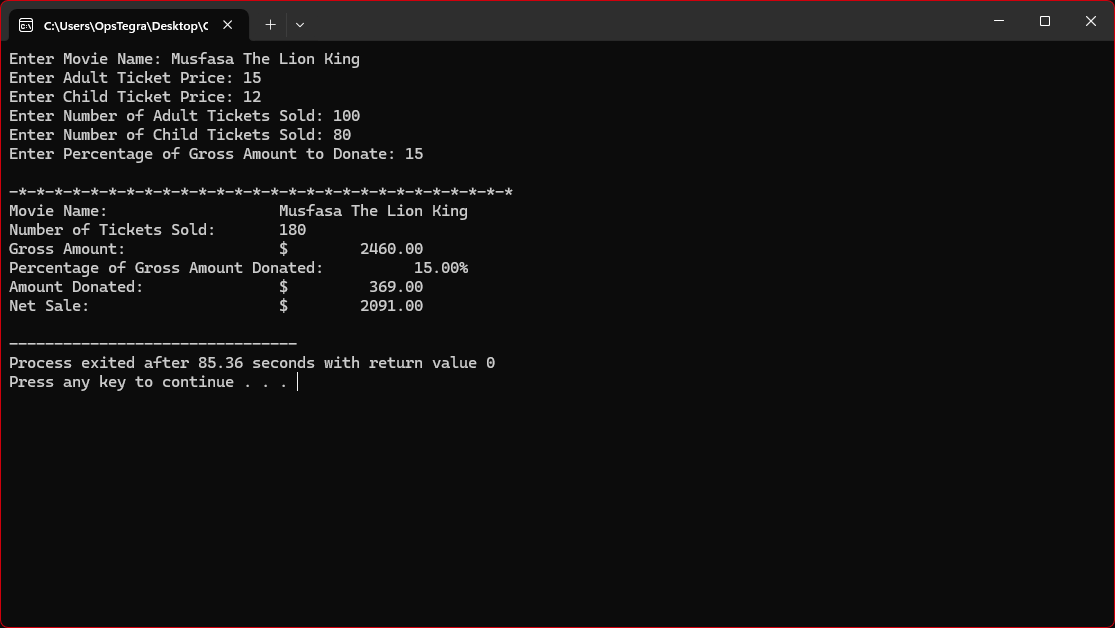
cout << left << setw(30) << "Amount Donated:" << "$" << right << setw(15) << fixed << setprecision(2) << amountDonated << endl;

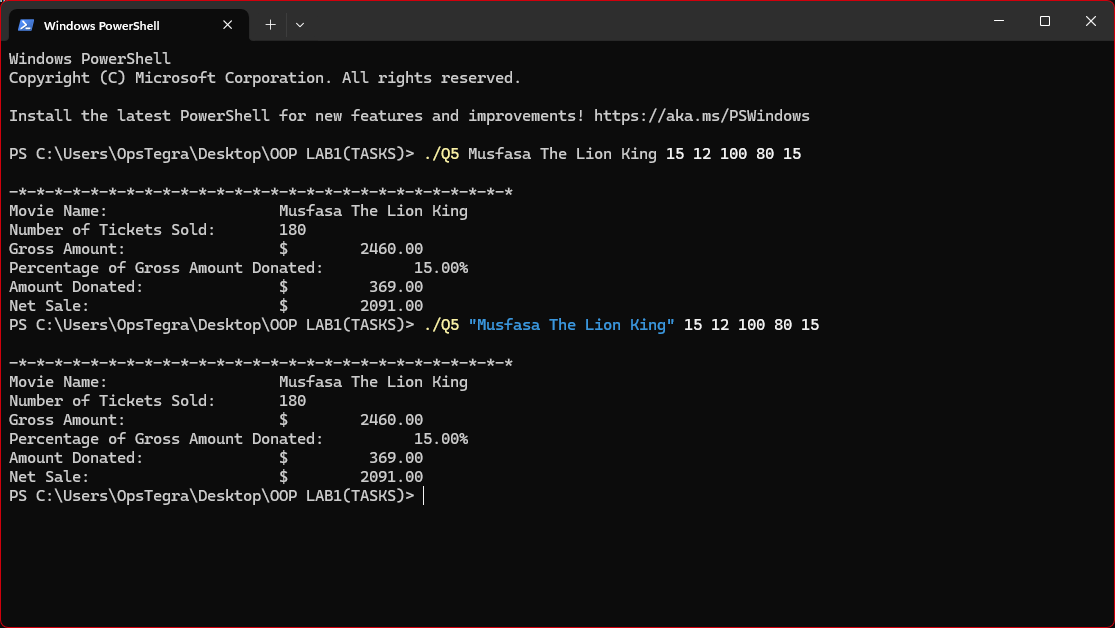
cout << left << setw(30) << "Net Sale:" << "$" << right << setw(15) << fixed << setprecision(2) << netSale << endl;

return 0;

}

**Result:**





**QUESTION NO.6**

**Q6:Write a program that reads a student name followed by five test scores. The program should output the student’s name, the five test scores, and the average test score. Output the average test score with two decimal places.**

**Input:**

**Andrew Miller 87.50 89 65.75 37 98.50**

**Output:**

**Student name: Andrew Miller**

**Test scores: 87.50 89.00 65.75 37.00 98.50**

**Average test score: 75.55**

**Solution:**

#include <iostream>

#include <string>

using namespace std;

int main(int argc, char \*argv[]) {

// ATTRIBUTES

string name = "";

double marks[5] = {0, 0, 0, 0, 0};

double sum = 0;

if (argc < 7) {

// FOR INTERACTIVE MODE

cout << "Input your name: ";

getline(cin, name);

cout << "==== Now enter marks for your 5 subjects ====" << endl;

for (int i = 0; i < 5; i++) {

cout << "Enter marks for Subject " << i + 1 << ": ";

cin >> marks[i];

sum += marks[i];

}

}

else {

// FOR COMMAND-LINE ARGUMENT

for (int i = 1; i < argc - 5; i++) {

name += argv[i];

if (i != argc - 6) name += " "; // Adds space between words in name

}

for (int i = 0; i < 5; i++) {

marks[i] = atof(argv[argc - 5 + i]);

sum += marks[i];

}

}

// CALCULATING AVERAGE

double avg = sum / 5;

// OUTPUT

cout << "Student Details!" << endl;

cout << "Student Name: " << name << endl;

cout << "Test Scores: ";

for (int i = 0; i < 5; i++) {

cout << marks[i] << " ";

}

cout << endl;

cout << "Average Test Score: " << avg << endl;

return 0;

}

**Result:**

