**Review C++ exercises – Selected Model Answers**

These answers are just suggestions – your solution may be different! Please ask you tutor if you have any questions or want further explanation.

1.

#include <string>

#include <iostream>

using namespace std;

int main() {

/\* The program should ask the user to enter

the rectangle width and height,

and output the area (width \* height) and perimeter

(2\* width + 2 \* height).\*/

double width, height;

char input;

cout << "Width? ";

cin >> width;

cout << "Height? ";

cin >> height;

double area = width \* height;

double perimeter = 2 \* (width + height);

cout << "Area is " << area << " Perimeter is " << perimeter << endl;

cout << "Press any key to exit" << endl;

cin >> input;

}

2.

int main() {

/\* Write a C++ program which asks the user to input 10 integer values,

and then prints out the maximum of the 10 values.T

here is no need to store the values. \*/

int input, max = -9999;

for (int i = 1; i <= 10; i++) {

cout << "Please enter integer " << i << ": ";

cin >> input;

if (input > max) {

max = input;

}

}

cout << "Maximum number input is " << max << endl;

cout << "Press any key to exit" << endl;

cin >> input;

}

3.

int main() {

int option;

option = showMenu();

while (option != 4) {

switch (option) {

case 1:

cout << "What a lovely sunny day" << endl;

break;

case 2:

cout << "Hope the sun comes back out soon" << endl;

break;

case 3:

cout << "Don't forget your umbrella" << endl;

break;

default:

cout << "Invalid option" << endl;

}

option = showMenu();

}

cout << "Goodbye!";

cin >> option;

}

4.

#include <string>

#include <iostream>

using namespace std;

struct PlayerData {

float health;

int score;

int livesLeft;

string id;

};

PlayerData \* inputData() {

PlayerData \* p = new PlayerData;

cout << "Please enter health (0-1.0), score, lives left and id: ";

cin >> p->health;

cin >> p->score;

cin >> p->livesLeft;

cin >> p->id;

return p;

}

void outputData(PlayerData \* p) {

cout << p->health << " , " << p->score << ", " << p->livesLeft << ", " << p->id << endl;

}

int main() {

PlayerData \* thePlayer;

thePlayer = inputData();

outputData(thePlayer);

delete(thePlayer);

int input;

cin >> input;

}