|  |  |
| --- | --- |
| Edith Cowan University School of Science |  |

Module 4

Performing Loops

**Related Objectives**

* Learn about the while loop
* Write some typical loops
* Learn to avoid common pitfalls with loops
* Accumulate totals
* Learn about the for loop
* Learn about pre-test and post-test loops

**Activity**

1. Write a program which uses the loop outlined below to validate a user data entry. The input and output requirements are shown below. Get your tutors feedback once you have completed this.

while (idNum < LOW\_NUM || idNum > HIGH\_NUM) {

cout << “Invalid ID number” << endl;

cout << “Please enter a number from “ << LOW\_NUM << “ through “ << HIGH\_NUM << “: “;

cin >> idNum;

}

Output:

Enter an ID number between 111 and 999 inclusive: 102

Invalid ID number

Enter an ID number between 111 through 999: 110

Invalid ID number

Enter an ID number between 111 through 999: 2222

Invalid ID number

Enter an ID number between 111 through 999: 456

Thank you. Your valid ID is 456

1. Complete the following code, compile and run. On successful execution the program will output something similar to the following output example.

for (count = 2, nextSessionMonth = aClient.firstSessionMonth, nextSessionYear = aClient.firstSessionYear; count <= aClient.numSessions; ++count) {

nextSessionMonth += 1;

if (nextSessionMonth > MONTHS\_IN\_YEAR) {

nextSessionMonth -= MONTHS\_IN\_YEAR);

++nextSessionYear;

}

cout << “Session #” << count << “ is on: “ << aClient.firstSessionDay << “/” << nextSessionMonth << “/” << nextSessionYear << endl;

}

Output:

Enter an ID number between 1000 and 9999 inclusive: 22

Invalid ID number

Enter an ID number from 1000 through 9999: 2518

Enter number of sessions client needs: 12

We guarantee that no more than 4 will be necessary.

Please re-enter the number of sessions: 4

Enter the day of the first session: 7

Enter the month of the first session: 11

Enter the year of the first session: 2015

Client #2518

First session is on the following day: 7/11/2015

Session #2 is on: 7/12/2015

Session #3 is on: 7/1/2016

Session #4 is on: 7/2/2016

1. Based on the following C++ program (which contains errors), you are required to debug and correct the code. On successful execution the program will output something similar to the output example below.

/\* This is a simple program to practice the use of nested for loops and structs.

\* The program uses structs of 2 arrays to generate 2x2 matrices.

\* The following contain a number of errors. Use this to practice debugging C++ programs \*/

#include <iostream>

using namespace std;

struct Matrix {

int column[10], row[10];

};

int main() {

Matrix myMatrix;

cout << “This is a simple example of nested loops, which generate a 2x2 matrix.” << endl;

for (i = 0; i < 10; i++) {

myMatrix.column[i] = i;

cout << “Iteration i = “ << i << endl;

for (j = 0; j < 11; --j) {

myMatrix.row[j] = j;

cout << “Iteration j = “ << j << “, “ << myMatrix.column[i] << “, “ << myMatrix.row[j] << endl;

}

system(“pause”);

return 0;

}

Output:

This is a simple example of nested loops, which generate a 2x2 matrix.

Iteration i = 0

Iteration j = 0, 0, 0

Iteration j = 1, 0, 1

Iteration j = 2, 0, 2

Iteration j = 3, 0, 3

Iteration j = 4, 0, 4

Iteration j = 5, 0, 5

Iteration j = 6, 0, 6

Iteration j = 7, 0, 7

Iteration j = 8, 0, 8

Iteration j = 9, 0, 9

Iteration i = 1

Iteration j = 0, 1, 0

Iteration j = 1, 1, 1

Iteration j = 2, 1, 2

Iteration j = 3, 1, 3

Iteration j = 4, 1, 4

Iteration j = 5, 1, 5

Iteration j = 6, 1, 6

Iteration j = 7, 1, 7

Iteration j = 8, 1, 8

Iteration j = 9, 1, 9

Iteration i = 2

Iteration j = 0, 2, 0

1. Write a program that crates an array of the following integers: [1,2,3,4,3,2,1] and performs the following operations using loops and decision statements:
   1. Print out the array elements.
   2. Sum the elements of the array and print out the total.
   3. Ask the user to enter a number (between 1 and 4 inclusive) and then find and print how many times that number occurs in the array.
   4. Replace all elements greater than 2 with 0.
   5. Shift the array by one element to the right, so that element 0 becomes element 1, element 1 becomes element 2… and the last element becomes element 0.
   6. Change your array elements to another set of integers and make sure your code still works.