**Practical 6 – refer to Topics 11 and 12**

**Part A (Understanding Concepts)**

1. If originally a = 4, b = -1, and c = 2, trace the following program statements by filling a table with the following format:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Statement Part | Condition | a | b | c |
|  |  |  |  |  |
|  |  |  |  |  |

1. if ( a > 0 )

a = 0;

if ( b > 0 )

b = 0

if ( c > 0 )

c = 0;

(b) if ( a > 0 )

a = 0;

else if ( b > 0 )

b = 0;

else if ( c > 0 )

c = 0;

1. Consider the following flowchart:
2. Which multi-way selection implementation would you use: nested if statements or a switch statement? Explain why.
3. Write the equivalent C/C++ statements.

true

true

false

pH > 7

pH < 12

pH is 7

pH > 2

Display “Neutral”

Display “Acidic”

Display “Very Acidic”

Display “Very Alkaline”

Display “Alkaline”

false

false

false

true

true

Multi-way selection

#include <iostream>  
if (ph >7)

{  
if (ph < 12)  
cout << “alkaline”;

Else if ( ph >12)  
cout << “very alkaline”

}  
else if (ph <7)  
{  
if (ph == 7)  
cout << “neutral”;  
else if (ph >2)  
cout << “acidic”  
else  
cout << “very acidic”;

1. Consider the following decision table.
2. Which multi-way selection implementation would you use: nested if statements or a switch statement? Explain why.
3. Write the C/C++ statements to display the appropriate brightness in lumens of a light bulb with a given wattage. If the wattage for a light bulb is not known, display a message “Unknown wattage”. Use switch and nested if statements, if appropriate.

|  |  |
| --- | --- |
| **Light Bulb Wattage (in watts)** | **Brightness (in lumens)** |
| 15 | 125 |
| 25 | 215 |
| 40 | 500 |
| 60 | 880 |
| 75 | 1000 |
| 100 | 1675 |
|  |  |

#include < iostream>  
using namespace std;

Int main(void)

{

Int watts;

Cout << “enter light bulb wattage (in watts): “;

Cin >> watts;

Switch (watts)

{  
Case 15: cout << “125 lumens\n”;

Break;  
Case 25: cout << “215 lumens\n”;

Break;  
Case 40: cout << “500 lumens\n”;

Break;

Case 60: cout << “880 lumens\n”;

Break;

Case 75: cout << “1000 lumens\n”;

Break;  
case 100:cout << “1675 lumes\n”;  
 break  
default: cout << “unknown wattage\n”;  
 break;

}

}

1. Consider the following decision table. (Assume wind speed is given as an integer.)
2. Is the switch statement suitable to implement the decision table? Why or why not?
3. Write nested if statements to test the wind speed and display the appropriate category. Start with the test for wind speed below 25.
4. Write nested if statements to test the wind speed and display the appropriate category. Start with the test for wind speed above 72.

|  |  |
| --- | --- |
| **Wind Speed** | **Category** |
| Below 25 | Not a strong wind |
| 25 - 38 | Strong wind |
| 39 - 54 | Gale |
| 55 - 72 | Whole gale |
| Above 72 | Hurricane |

If (wind speed < 25)  
 cout << “Not strong”;

Else if (wind speed <= 38)  
 cout <<“strong wind”;  
Else if (ws <= Gale <= 54)

Cout<<“gale”;  
Else if (wind speed <= 72)   
 cout << “whole gale”;  
Else  
 Cout<< “hurricane”;  
  
OR   
  
if (wind speec >= 72)  
cout <<”hurricane”;  
else if (wind speed >=55)  
cout << “whole gale”;  
else if (wind speed >=39)  
cout << “gale”;  
else if (wind speed >= 25)  
cout << “strong wind”;  
else   
cout<< “ not strong wind”;

1. Trace the following program statements by filling a table with the following format. Use an ‘◘’ to indicate a space in the output.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *variable-1* | *variable-2* | *. . .* | *variable-n* | *condition* | output |
|  |  |  |  |  |  |
|  |  |  |  |  |

1. i = 0;

while ( i <= 5 )

{

cout << setw(3) << i << setw(3) << 10 - i << endl;

i ++;

}

|  |  |  |
| --- | --- | --- |
| i | i <= 5 | OUTPUT |
| 0 | TRUE | ◘◘0◘10 |
| 1 | TRUE | ◘◘1◘9 |
| 2 | TRUE | ◘◘2◘8 |
| 3 | TRUE | ◘◘3◘7 |
| 4 | TRUE | ◘◘4◘6 |
| 5 | TRUE | ◘◘5◘5 |
| 6 | FALSE |  |

1. cin >> n; /\* assume value entered is 30 \*/

v = 5;

while ( v < n )

{

cout << setw(3) << v;

v += 5;

}

cout << endl;

|  |  |  |  |
| --- | --- | --- | --- |
| n | v | V < n | OUTPUT |
| 30 | 5 | TRUE | ◘◘5◘10◘15◘20◘25 |
| 30 | 10 | TRUE |  |
| 30 | 15 | TRUE |  |
| 30 | 20 | TRUE |  |
| 30 | 25 | TRUE |  |
| 30 | 30 | FALSE |  |

1. cin >> start; /\* assume value entered is 3 \*/

cin >> end; /\* assume value entered is 12 \*/

cin >> increment; /\* assume value entered is 2 \*/

side = start;

while ( side <= end )

{

area = side \* side;

cout << setw(2) << side << setw(3) << area << endl;

side += increment;

}

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start | End | Increment | Side | Area | Side <= end | output |
| 3 | 12 | 2 | 3 | 9 | TRUE | ◘3◘◘9 |
| 3 | 12 | 2 | 5 | 25 | TRUE | ◘5◘25 |
| 3 | 12 | 2 | 6 | 49 | TRUE | ◘7◘49 |
| 3 | 12 | 2 | 9 | 81 | TRUE | ◘9◘81 |
| 3 | 12 | 2 | 11 | 121 | TRUE | 11121 |
| 3 | 12 | 2 | 13 |  | FALSE |  |

1. Rewrite the program fragments in question 5 using a for statement instead of the while statement.

(a) for (I = 0, ; I <= 5 ; I += 5)

cout << setw(3) << i << setw(3) << 10 - i << endl;

b)   
cin >> n;

for (v=5; v <n; v+=5)

cout << setw(3) << v;

cout << endl;

c)  
  
 cin >> start; /\* assume value entered is 3 \*/

cin >> end; /\* assume value entered is 12 \*/

cin >> increment; /\* assume value entered is 2 \*/

for ( side = start; side <= end; side += increment)  
{

area = side \* side;

cout << setw(2) << side << setw(3) << area << endl;

}

1. Write the for statement to produce the following output:
2. -100 -90 -80 -70 -60 -50 -40 -30 -20 -10 0 10 20 30

z  
for (int I =- 100 ; I <= 30 ; i+= 10)

cout << I << “ “;

1. 0 1
2. 2
3. 4
4. 8
5. 16
6. 32
7. 64

for (I = 0; I <= 6; i++)

cout << I “\t” << pow(2.0, i) << endl;

8. The flowchart below describes a program that computes the average speed, given the time taken and the distance travelled. Extend the flowchart so that the program computes the average speed for 10 different sets of distances and times.

Start

Read distance, time

ave\_speed = distance / time

Display ave\_speed

Stop

#include<iostrem>  
using namespace std;  
  
int main(void  
{  
double distance, time, ave\_speed;  
  
for (int count = 0; count < 10; count++)  
{

Cout << “enter distance and time”;  
cin >> distance >> time;  
  
ave\_speed = distance / time;  
  
cout << “average speed =” << ave\_speed<< endl;  
}  
return 0;

}

**Part B (Programming Exercises)**

1. Implement the extended flowchart for Part A question 8.
2. #include <iostream>
3. using namespace std;
4. int main()
5. {
6. double distance, time, ave\_speed;
7. int i;
8. for (i = 0; i < 10; i++)
9. {
10. cout << "Enter distance & time";
11. cin >> distance >> time;
12. ave\_speed = distance / time;
13. cout << "Average speed is: " << ave\_speed << endl;
14. }
15. }
16. Write a program that implements the following flowchart.

c = b

c = a

b = a + 1

c = c +1

b = a - 1

c = 2 \* a

Display a, b, c a,b,c

a > 0

b > 0

true

true

false

false

Stop

Start

Read a,b

#include <iostream>

using namespace std;

int main()

{

int a, b, c;

cout << "Enter a & b ";

cin >> a >> b;

if (a > 0)

{

if (b > 0)

{

c = a;

b = a + 1;

}

else

c = b;

c = c + 1;

}

else

{

b = a - 1;

c = 2 \* a;

}

cout << "a= " << a << "b=" << b << "c=" << c << endl;

return 0;

}

1. Write a program to allow a user to enter a measurement in centimeters and then converts and displays the measurement in inches. One centimeter equals 0.3937inch.

#include <iostream>

#define CM\_TO\_INCH 0.3957

using namespace std;

1. int main()
2. {
3. double cm, inch;
4. int i;
5. for (i = 0; i < 6; i++)
6. {
7. cout << "Enter measurement in cm:";
8. cin >> cm;
9. inch = cm \* CM\_TO\_INCH;
10. cout << " The measurement in inches is " << inch << endl;
11. }
12. return 0;
13. }
14. Modify the program in question 3 to allow the user to perform the conversion for 6 different measurements.
15. Write a program to display a centimeters-to-inches conversion table, starting with 1 centimeter until 20 centimeters in steps of 1 centimeter. Include a table heading. Use integer type for centimeters.
16. #include <iostream>
17. #define CM\_TO\_INCH 0.3957
18. using namespace std;
19. int main()
20. {
21. double cm, inch;
23. cout << " CM \tIN" << endl;
24. for (cm = 1; cm <= 20; cm++)
25. {
26. inch = cm \* CM\_TO\_INCH;
27. cout << cm << "\t" << inch << endl;
28. }
29. return 0;
30. }

1. Modify the program in question 5 to allow the user to enter the starting, ending, and increment values for the centimeters.
2. #include <iostream>
3. #define CM\_TO\_INCH 0.3957
4. using namespace std;
5. int main()
6. {
7. double cm, inch;
8. int start, end, increment;
10. cout << " Enter starting cm ";
11. cin >> start;
12. cout << " Enter ending cm";
13. cin >> end;
14. cout << " Enter increment:";
15. cin >> increment;

**Part C (Self-Review / Revision)**

1. What is a loop?  
   repetition of steps in a program
2. What is an infinite loop?

Loop that never stops

1. What is a loop iteration?  
   loop iteration is one execution of the loop body
2. What is a counter-controlled loop?   
   loop where its repetition is managed by a loop control variable called a counter which represent a loop count
3. What are the 3 things that must be done for a counter-controlled loop variable?  
   the 3 things that must be done for a counter-controlled loop variables are:  
   -initialization of the variable before the first loop iteration

**-testing of its value before start of each loop iteration**

**-updating of its value at the end of each loop iteration**

1. What is a pretest loop? What is a post-test loop?

Pre test Loop where we test the condition for stopping the loop before each iteration of the loop

**Post test loop is a loop where we test the condition for stopping the loop before each iteration of the loop**

1. What are two statements in C/C++ for implementing a pretest counter-controlled loop?

While & for statement

**Part D (Practice Exercises)**

1. Write a program to generate the multiplication table for a number entered by the user. For example, if the user enters 6, the program displays:

1 x 6 = 6

2 x 6 = 12

. . .

12 x 6 = 72

#include <iostream>

# include <iomanip>

using namespace std;

int main()

{

int n, i;

cout << "Enter number";

cin >> n;

for (i = 1; i <= 12; i++)

{

cout << setw(2) << i << " x " << setw(2) << n << " = " << setw(2) << i \* n << endl;

}

return 0;

}