**Practical 5 – refer to Topics 09 and 10**

**Part A (Understanding Concepts)**

1. For the following flowcharts, write the equivalent pseudocode and the complete C/C++ program.

|  |  |
| --- | --- |
| Start  Read three numbers  Find the sum of the three numbers  Display the sum  Stop | Start  Read num1, num2  Stop  num1>num2  Display “Second number is bigger”  Display “First number is bigger”  false  true |
| (a) Sequence | (b) Two-way selection |
| START  Read 3 numbers  Find sum of 3 numbers  Display sum of 3 numbers  STOP  (a) #include <iostream>  using namespace std;  int main(void)  {  //read three numbers  int num1, num2, num3;  cout << "Enter 3 numbers: ";  cin >> num1 >> num2 >> num3;  //find sum of 3 numbers  int sum = num1 + num2 + num3;  //display sum  cout << "The sum of " << num1 << ", " << num2 << " and "  << num3 << " is " << sum << ".\n";  return 0;  }  OR |  |

(b)

START

Read num1, num2

If num1 > num2

Display “first number is bigger”

Else

Display “Second number is bigger”

STOP

#include <iostream>

using namespace std;

int main(void)

{

//read num1, num2

int num1, num2;

cout << "Enter 2 numbers: ";

cin >> num1 >> num2;

//if num1 > num2

// display "first number is bigger"

//else

// display " sec no is bigger"

if (num1 > num2)

cout << "first number is bigger";

else

cout << "second number is bigger";

return 0;

}

1. Suppose that a = -2, b = 5, c = 0. Evaluate the following expressions. Show the steps in the evaluation as illustrated in this example. Apply short-circuit evaluation where it is possible.

Example:

a + 2 + c && b > c (Short-circuit evaluation possible)

With short-circuit evaluation

a + 2 + c && b > c

=> -2 + 2 + 0 && 5 > 0

=> 0 + 0 && 5 > 0

=> 0 && 5 > 0

=> F && 5 > 0

=> F

1. a + b < c + 2

-2 + 5 < 0 + 2  
3 < 2   
FALSE

1. 2 + a \* b + 5 < c \* 2 / 3

2 + -2 \* 5 + 5 < 0 \* 2 / 3

First operator +, multiplier > addition

2 + -10 + 5 < 0 / 3

2 + -10 + 5 < 0

-8 + 5 < 0

-3 < 0  
**true**

1. b == 5 && 3 \* b / 4 % b (if left is true, need test right)

5 == 5 && 3 \* 5 / 4 % 5

True && 15 / 4 % 5

True && 3 % 5

True && 3 (zero represents false; any non-zero value represent true)

True && True

**True**

1. c + 8 >= b || a \* 7

0 + 8 >= 5 || -2 \* 7

8 >= 5 || -2 \* 7

True || -2 \* 7

**True**

1. !(a <= b) && (a \* b != c)  
   ! (-2 <= 5) && (-2 \* 5 ! = 0)  
   ! T && T  
   F && T  
   FALSE
2. Given the following statements, what is the output if number is:
3. 17
4. 170

inRange = (-5 <= number && number <= 100);

if ( inRange )

cout << "In the range";

else

cout << "Not in the range";

(a) bool inRange = (-5 <= number && number <= 100);

-5 <= 17 && 17 <= 100

True && True

True

cout << "In the range";

(b) bool inRange = (-5 <= number && number <= 100);

-5 <= 170 && 170 <= 100

True && False

False

cout << "Not in the range";

1. If originally a = 4, b = -1, and c = 2, desk-check the following program statements by filling in the table below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Statement Part | Condition | a | b | c |
|  |  | 4 | -1 | 2 |
| if ( a != 0 ) | 4 != 0 -> true | 4 | -1 | 2 |
| b = b + 1; |  | 4 | 0 | 2 |
| C = 0; |  | 4 | 0 | 0 |

1. if ( a != 0 )

b = b + 1;

else

b = b - 1;

c = 0;

if ( a != 0 )

{

b = b + 1;

}

else

{

b = b - 1;

}

c = 0;

1. if ( a == 0 )

b = b + 1;

c = 0;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Statement Part | Condition | a | b | c |
|  |  | 4 | -1 | 2 |
| if ( a == 0 ) | 4 = 0 is false | 4 | -1 | 2 |
| C = 0; |  | 4 | -1 | 0 |

1. if ( a < b )

b = a;

else

{

a = b;

c = b;

}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Statement Part | Condition | a | b | c |
|  |  | 4 | -1 | 2 |
| if ( a < b ) | 4 <-1 is false | 4 | -1 | 2 |
| Else  {  A=b;  C=b;  } |  | -1 | -1 | -1 |

1. Simplify the following conditions by applying DeMorgan’s rule.(T10 S29)
2. ! ( age >= 18 && status == 'C' )

! (age >= 18) || ! (status == ‘C’)

(age < 18) || (status != ‘C’)

1. ! ( unemployed || dependents > 4 )

! unemployed && ! (dependents > 4 )  
! unemployed && ( dependents <= 4 )

1. Write the function definitions for each of the following functions based on the function prototypes and descriptions given:
2. int isEven (int num);

The function has a parameter of type int and returns true (1) if the integer is an even number and false (0) otherwise.

bool isEven (int num);

{

If ( num % 2 == 0 )

return true;

else

return false;

}

Bool isEven (int num)

{

Return ( num % 2 == 0 );

}

1. int isDigit (char ch);

The function has a parameter of type char and returns true (1) if the character is a digit (‘0’-‘9’) and false (0) otherwise.

bool isDigit ( char ch);

{

Return (‘0’ <= ch 7& ch <= ‘9’ )

}  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#include <iostream>

using namespace std;

bool isDigit(char ch);

int main(void)

{

char ch;

cout << "Enter a character: ";

cin >> ch;

bool flag = isDigit(ch);

if (flag == true)

cout << ch << " is a digit.\n";

else

cout << ch << "is NOT a digit.\n";

return 0;

}

bool isDigit(char ch)

{

return('0' <= ch && ch <= '9');

}

1. (a) Draw a flowchart for a program that asks a user to enter two characters into two variables named ch1 and ch2. The program checks if ch2 is lower than ch1 in the alphabetical ordering. If yes, the program swaps the characters. For example, if ch1 is ‘z’ and ch2 is ‘g’, then the ch1 should become ‘g’ and ch2 should become ‘z’. Finally the program displays the characters in ch1 and ch2 in that order.

Start

Read ch1, ch2

Ch2<ch1

Temp = ch1

Ch1 = ch2

Ch2 = temp

Display ch1, ch2

stop

1. Write the corresponding if or if-else statement.

If (ch2 < ch1)

{

Temp = ch1;

Ch1 = ch2;

Ch2 = temp;

}

1. (a) Draw a flowchart for a program that allows the user to find the area of a rectange or the area of a circle after prompting the user to choose ‘R’ or ‘r’ for rectangle and ‘C’ or ‘c’ for circle into a variable named choice. The area of a rectangle is length x width and the area of a circle is π x radius2 where π is 3.14159. Assume the user enters correct input.

Start

Read choice

false true

Choice is ‘R’ or ‘r’

Read length, width

Read radius

Area = length x width

Area = pie x radius^2

Display area of rectangle

Display area of circle

stop

(b) Write the corresponding if or if-else statement.

If ( choice == ‘R’ || choice == ‘r’)

{

cout << “ Enter length & Width”;

cin >> length >> width;

area = length \* width;

cout << “area is “ << area << endl;

}

Else

{  
cout << “ Enter radius” ;

Cin >> radius;

Area = pie \* radius ^2;

Cout << “area of circle is” << area << endl;

}

**Part B (Programming Exercises)**

1. Write and test the program for Part A question 7.  
   #include <iostream>

using namespace std;

int main(void)

{

char ch1, ch2;

cout << "Enter a character: ";

cin >> ch1 >> ch2;

if (ch2 < ch1)

{

char temp = ch1;

ch1 = ch2;

ch2 = temp;

}

cout << "the entered characters are "

<< ch1 << " and " << ch2 << ".\n";

return 0;

}

1. Write and test the program for Part A question 8. Use the pow function in your answer. Use data type double.

#include <iostream>

#include <cmath>

#define PI 3.14159

using namespace std;

int main(void)

{

char choice;

cout << "Do you want to compute area of rectangle (R) or circle (C): ";

cin >> choice;

if (choice == 'R' || choice == 'r')

{

double length, width;

cout << "Enter length and width: ";

cin >> length >> width;

double area = length \* width;

cout << "The area of the rectangle is " << area << ".\n";

}

else

{

double radius;

cout << "Enter radius: ";

cin >> radius;

double area = PI \* pow((double)radius, 2.0);

cout << "The area of the circle is " << area << ".\n";

}

return 0;

}

1. Write a program that accepts a number from the user and displays a message to indicate whether the number is even or odd. Use the function you wrote in Part A question 6(a).
2. #include <cmath>
3. #define PI 3.14159
4. using namespace std;
5. bool isEven(int num);
6. int main(void)
7. {
8. int num;
9. cout << "Enter a number: ";
10. cin >> num;
11. if (isEven(num))
12. cout << "An even number.";
13. else
14. cout << "An odd number. ";
15. return 0;
16. }
17. bool isEven(int num)
18. {
19. return (num % 2 == 0);
21. }
22. Write a program that accepts a character from the user and displays a message to indicate whether the character is a digit or not. Use the function you wrote in part A question 6(b).
23. #include <iostream>
24. #include <cmath>
25. #define PI 3.14159
26. using namespace std;
27. bool isDigit(char ch);
28. int main(void)
29. {
30. char ch;
31. cout << "Enter a character: ";
32. cin >> ch;
33. if (isDigit(ch))
34. cout << " is a digit.\n";
35. else
36. cout << "is NOT a digit.\n";
37. return 0;
38. }
39. bool isDigit(char ch)
40. {
41. return('0' <= ch && ch <= '9');
42. }
43. Write a program that determines if a person may be accepted as a potential astronaut candidate by asking the user some personal details. To be accepted, the person’s age must be between 35 and 45, the weight between 50.0kg and 80.0kg, and the person must be a non-smoker.

A sample run of the program is as follows:

Want to be an astronaut?

Please answer the following questions:

What is your age? 37

What is your weight (kg)? 65.5

Are you a smoker (y or n)? y

Sorry we cannot accept you as an astronaut.

#include <iostream>

using namespace std;

int main(void)

{

cout << "Bro, wanna be an astronot?\n";

cout << "Pls answer these question:\n";

int age;

cout << "wot is ur age? ";

cin >> age;

double weight;

cout << "how fat r u (kiloton) ?";

cin >> weight;

char smoker;

cout << "u smoke pot? yes or no? ";

cin >> smoker;

if ((35 <= age && age <= 45) &&

(50.0 <= weight && weight <= 80.0) &&

(smoker == 'N' || smoker == 'n'))

cout << "HURRAY! "

<< "u r qualified m8.\n";

else

cout << "sorry you suck.\n";

return 0;

}

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

1. What are the 3 control structures in structured programming languages?  
   Sequence, selection, repetition
2. What is the purpose of a flowchart? What are the symbols used in a flowchart and their meanings?  
   Use as a program design tool,  
   Use to design the algorithm for the solution to a problem just like pseudocode   
   It represent the logical flow of a program or the program logic in graphical form
3. How does C/C++ represent a true and a false value?

True non-zero  
False 0  
  
Build-in Boolean type(in C++ only)

1. What is short-circuit evaluation?   
   True || Anything = True  
   False && Anything = False
2. How are characters stored in the computer?  
   in the Computer using a coding system.  
   a common coding system is ASCII
3. What does DeMorgan’s rule say?  
   !(expr1&&expr2) is written as (!expr1 || !expr2)

! (expr1||expr2) is written as (!expr1&& !expr2)

1. How does C/C++ implement two-way selection? one-way selection?  
   2 ways with if-else statement  
   1 way with a if statement
2. What is a compound statement?  
   a compound statement is a group of statements enclosed in braces

**Part D (Practice Exercises)**

1. A water supply company charges $1.30 for every 1000 gallons of water used. The amount of water consumed is computed based on the beginning and ending meter readings entered in gallons.

To encourage consumers to save water resources, the company gives 100% discount (i.e. sets water charge to $0.0) if the amount of water consumed is less than 800 gallons.

Design (draw a flowchart ) and write the program to compute the water charge. Use three functions as follows:

* read\_meters:

to read and return the beginning and ending meter readings.

* compute\_charge:

to compute and return the water charge given the beginning and ending meter readings.

* display\_charge:

to display the water charge. If the charge is $0.0, the following message should be displayed as well: “Thank you for saving precious water resource”.

*Note: Define the charge per 1000 gallons as a defined constant.*#include <iostream>

#include <iomanip>

#define GALLON 1.30

using namespace std;

void read\_meter(int\*begin, int\*end);

double compute\_charge(int begin, int end);

void display\_charge(double charge);

int main(void)

{

int begin, end;

double charge;

read\_meter(&begin, &end);

charge = compute\_charge(begin, end);

display\_charge(charge);

return 0;

}

void read\_meter(int\* begin, int\*end)

{

cout << "Enter the begin and end of meter reading: ";

cin >> \*begin >> \*end;

}

double compute\_charge(int begin, int end)

{

int consumed;

double charge;

consumed = end - begin;

if (consumed < 800)

charge = 0.0;

else

charge = consumed / 1000.0 \* GALLON;

return charge;

}

void display\_charge(double charge)

{

cout << fixed << setprecision(2) << "Water bill is : " << charge << endl;

if (charge == 0)

cout << "Thanks for saving precious H20!" << endl;

}