Condition Related Problems

(Total 15 questions)

SL	Problem statement	Difficulty levels
1.	Program that will decide whether a number is positive or not. 	*
2.	Program that will decide whether a number is even or odd. 	*
3.	Program that will take an integer of length one from the terminal and then display the digit in English. 	*
4.	Program that will check whether a triangle is valid or not, when the three angles (angle value should be such that, 0 < value < 180) of the triangle are entered through the keyboard. [**Hint:** A triangle is valid if the sum of all the three angles is equal to 180 degrees.] 	*

Problem 1:

Sample input	Sample output
100	Positive
-11.11	Negative
0	Positive

Problem 2:

Sample input	Sample output
50	Even
-77	Odd
0	Even

Problem 3:

Sample input	Sample output
9	nine
0	zero

Problem 4:

Sample input			Sample output
90	45	45	Yes
30	110	40	Yes
160	20	30	No
0	180	0	No

5.	Program that will read from the console a random positive nonzero number and determine if it is a power of 2.	**

Sample input	Sample output
1	Yes
512	Yes
1022	No

6.	Program that will read from the console a random number and check if it is a nonzero positive number. If the check is yes, it will determine if the number is a power of 2. If the check fails the program will check for two more cases. If the number is zero, the program will print "Zero is not a valid input". Else it will print "Negative input is not valid".	***

Sample input	Sample output
0	Zero is not a valid input
1	Yes
512	Yes
1022	No
-512	Negative input is not valid

7.	Program that will take two numbers **X** & **Y** as inputs and decide whether **X** is greater than/less than/equal to **Y.**	*

Sample input (X,Y)	Sample output
5 -10	5 is greater than -10
5 10	5 is less than 10
5 5	5 is equal to 5

8.	Program that will decide whether a year is leap year or not. Yes, if (Year % 4 == 0 && year % 100 != 0) ‖ (Year % 400 ==0)	*

Sample input	Sample output
2000	Yes
2004	Yes
2014	No

9.	Program that will categorize a single character that is entered at the terminal, whether it is an alphabet, a digit or a special character. (**Restriction:** Without math.h)	*

Sample input	Sample output
z	Alphabet
A	Alphabet
8	Digit
*	Special

10.	Program that will evaluate simple expressions of the form- <number1> <operator> <number2> ; where operators are (+, - , *, /) And if the operator is "/", then check if <number2> nonzero or not.	**

Sample input	Sample output
100 * 55.5	Multiplication: 5550
100 / -5.5	Division: -18.181818
100 / 0	Division: Zero as divisor is not valid!

11.	Program that will take the final score of a student in a particular subject as input and find his/her grade.	*

Marks	Letter Grade	Marks	Letter Grade	Marks	Letter Grade
90-100	A	70-73	C+	Less than 55	F
86-89	A-	66-69	C		
82-85	B+	62-65	C-		
78-81	B	58-61	D+		
74-77	B-	55-57	D		

Sample input	Sample output
91.5	Grade: A
50	Grade: F

12.	Program that will construct a menu for performing arithmetic operations. The user will give two real numbers **(a, b)** on which the arithmetic operations will be performed and an integer number (1 <= Choice <= 4) as a choice. Choice-1, 2, 3, 4 are for performing addition, subtraction, multiplication, division (quotient) respectively.	*

Sample input (a, b, Choice)	Sample output
5 10 3	Multiplication: 50
-5 10.5 4	Quotient: 0

13.	Program that will construct a menu for performing arithmetic operations. The user will give two real numbers **(a, b)** on which the arithmetic operations will be performed and an integer number (1 <= **Choice** <= 4) as a choice. Choice-1, 2, 3, 4 are for performing addition, subtraction, multiplication, division respectively. If Choice-4 is selected, again the program will ask for another choice (1 <= **Case** <=2), where Case-1, 2 evaluate quotient and reminder respectively.	**

Sample input	Sample output
5 10 3	Multiplication: 50
-5 10.5 4 1	Quotient: 0
-5 10.5 4 2	Reminder: -48

14.	Program that will construct a menu for performing arithmetic operations. The user will give two real numbers **(a, b)** on which the arithmetic operations will be performed and an integer number (1 <= **Choice** <= 4) as a choice. Choice-1, 2, 3, 4 are for performing addition, subtraction, multiplication, division respectively. If Choice-4 is selected, the program will check if **b** is nonzero. If the check is true, the program will ask for another choice (1 <= **Case** <=2), where Case-1, 2 evaluate quotient and reminder respectively. If the check is false, it will print an error message "Error: Divisor is zero" and halt.	***

Sample input	Sample output
5 10 3	Multiplication: 50
-5 10.5 4 2	Reminder: -48
-5 0 4	Error: Divisor is zero

15.	Program for "Guessing Game": Player-1 picks a number **X** and Player-2 has to guess that number within **N = 3** tries. For each wrong guess by Player-2, the program prints "Wrong, **N-1** Chance(s) Left!" If Player-2 successfully guesses the number, the program prints "Right, Player-2 wins!" and <u>stops allowing further tries (if any left)</u>. Otherwise after the completion of **N = 3** wrong tries, the program prints "Player-1 wins!" and halts. [**Restriction:** Without using loop/break/continue **Hint:** Use flag]	***

Sample input (X, n1, n2, n3)	Sample output
5 12 8 5	Wrong, 2 Chance(s) Left! Wrong, 1 Chance(s) Left! Right, Player-2 wins!
100 50 100	Wrong, 2 Chance(s) Left! Right, Player-2 wins!
20 12 8 5	Wrong, 2 Chance(s) Left! Wrong, 1 Chance(s) Left! Wrong, 0 Chance(s) Left! Player-1 wins!