



### Objective:

- To get a grip on problem solving based on sequence and selection structure.

### Devise Solution of the following problems using Pseudo Code

1. Input four numbers and display sum of these.
2. Input marks of five subjects of a student and calculate its average.
3. Input radius of circle and calculate area of circle.
4. A painter wants to know the amount of paint needed to paint only the four walls. The chosen paint covers 100 square feet per gallon.
5. Bob would like to know what percentage of his income his rent is. Write a solution that would calculate and print this percentage.
6. Jim is going to replant his lawn. He would like to know how much the sod will cost, given cost of cost. the sod and the length and width of two different lawn areas. Write a solution to calculate this
7. Write pseudo code to calculate circumference of circle.
8. Write a program that input a number and prints a multiplication table for the entered number up to 10.
9. Take input from user about temperature in Celsius and display on screen the equivalent Fahrenheit temperature.
10. Calculate the pay of an employee, given the hours worked and rate per hour.

### For Lecture # 2

1. Input age and print the message "eligible" if the age is valid enough to have a driving license otherwise print "not eligible".
2. Input an integer value representing the weekday (1, 2, ..., 7), and give the day of the week (Monday, Tuesday, ..., Sunday).
3. Write pseudo code, which takes input a number, and display its absolute value.
4. Check if marks are greater than 80 then display "good luck", otherwise display "better luck next time"
5. Check whether a number A is divisible by a number B.
6. Julio Cesar Chavez Mark VII is an interplanetary space boxer, who currently holds the championship belts for various weight categories on many different planets within our solar system. However, it is often difficult for him to recall what his "target weight" needs to be on earth in order to make the weight class on other planets. Write a pseudo code to help him keep track of this. It should ask him what his earth weight is, and to enter a number for the planet he wants to fight on. It should then

| # | Planet  | Relative gravity |
|---|---------|------------------|
| 1 | Venus   | 0.78             |
| 2 | Mars    | 0.39             |
| 3 | Jupiter | 2.65             |
| 4 | Saturn  | 1.17             |
| 5 | Uranus  | 1.05             |
| 6 | Neptune | 1.23             |



compute his weight on the destination planet based on the table below:  
So, for example, if Julio weighs 128 lbs. on earth, then he would weigh just under 50 lbs. on Mars, since Mars' gravity is 0.39 times earth's gravity. ( $128 * 0.39$  is 49.92)

**Sample Run 1**

Please enter your current earth weight: 128

I have information for the following planets:

- 1. Venus 2. Mars 3. Jupiter
- 4. Saturn 5. Uranus 6. Neptune

Which planet are you visiting? 2

Your weight would be 49.92 pounds on that planet.

7. One of the jobs that Joe Roberts has been given at work is to order special paper for a report for a board meeting. The paper comes in reams of 500 sheets. He always makes five more copies than the number of people that will be there. Joe wants to know how many reams of paper he needs for a meeting. He can order only whole, not partial, reams. Help Joe to develop a pseudo code that asks for number of people in meeting and report length in pages to determine the count of reams needed to purchase.

8. An admission charge for The Little Rep Theater varies according to the age of the person. Develop a solution to print the ticket charge given the age of the person. The charges are as follows:
- a. Over 55: \$10.00
  - b. 21-54: \$15.00
  - c. 13-20: \$10.00
  - d. 3-12: \$5.00
  - e. Under 3: Free

9. Input a 4 digit number and display each digit of it on separately on screen.

Sample Run:

Enter 4 digit Number: 5619  
5  
6  
1  
9

10. Determine the status of students (Safe/Dropped/Probation). Student is dropped if gpa is less than 1.7, on probation if gpa is greater than or equal to 1.7 but less than 2.0, otherwise student is safe.

11. Input from user three numbers and display on screen the largest, second largest and smallest number of them.

| Sample Run 1:   | Sample Run 2:   |
|---|---|
| Enter three Numbers: 100<br>34<br>923<br>Largest Number: 923<br>Second Largest Number: 100<br>Smallest Number: 34 | Enter three Numbers: 92<br>34<br>92<br>Largest Number: 92<br>Second Largest Number: 92<br>Smallest Number: 34 |

12. Calculate pay of an employee based on the hours worked. The input includes the employee total hours worked this week and his hourly pay rate. The employee is to be paid his basic wage for the first 40 hours worked and 50% more for all the hours above 40 (overtime pay). Output the regular pay, overtime pay, and total pay for the week. If the employee worked 40 hours or less then do not display any information about overtime pay.



**Sample Run 1**

Enter Employee Total Hours Worked: 50  
 Enter Employee Hourly Pay Rate: 10  
 Regular Pay: 400  
 Overtime Pay: 150  
 Total Pay: 550

**Sample Run 2**

Enter Employee Total Hours Worked: 20  
 Enter Employee Hourly Pay Rate: 5  
 Regular Pay: 100  
 Total Pay: 100

13. Input the time in 24-hour format and display time in 12 hours format.

14. Input marks of a student and display the letter grade and grade point.  
 a. Policy for letter grade is as follows:

| Percent Marks | Letter Grade | Grade Point |
|---------------|--------------|-------------|
| 0 - 49        | F            | 0           |
| 50 - 54       | D            | 1           |
| 55 - 57       | C-           | 1.7         |
| 58 - 60       | C            | 2           |
| 61 - 64       | C+           | 2.3         |
| 65 - 69       | B-           | 2.7         |
| 70 - 74       | B            | 3           |
| 75 - 79       | B+           | 3.3         |
| 80 - 84       | A-           | 3.7         |
| 85 - 100      | A            | 4           |

**15. Geometry Calculator**

Devise a pseudo code that displays the following menu:

Geometry Calculator

1. Calculate the Area of a Circle
2. Calculate the Area of a Rectangle
3. Calculate the Area of a Triangle

4. Quit

Enter your choice (1-4):

If the user enters 1, the program should ask for the radius of the circle and then display its area. Use the following formula:

$$\text{area} = \pi * r^2$$

Use 3.14159 for PI and the radius of the circle for r. If the user enters 2, the program should ask for the length and width of the rectangle and then display the rectangle's area. Use the following formula:

$$\text{area} = \text{length} * \text{width}$$

If the user enters 3 the program should ask for the length of the triangle's base and its height, and then display its area. Use the following formula:

$$\text{area} = \text{base} * \text{height} * .5$$

If the user enters 4, the program should end.

Display an error message if the user enters a number outside the range of 1 through 4 when selecting an item from the menu. Do not accept negative values for the circle's radius, the rectangle's length or width, or the triangle's base or height.



16. Input a 3 digit number and check whether it's a palindrome or not. Palindrome is a word, phrase, or sequence that reads the same backward as forward. For example, 717 is a palindrome number, "EYE" is palindrome word.

**Sample Run 1**

Enter a Three Digit Number: 797  
797 is Palindrome

**Sample Run 2**

Enter a Three Digit Number: 231  
231 is not Palindrome

- 1) Computer networking  
A top down Approach by  
James f. Kurose Keith W. Ross  
5<sup>th</sup>/6<sup>th</sup>
- 2) Data base system  
by  
Ramez Elmasri  
6<sup>th</sup> Edition  
Shankant B. Navathe



b/10

6-10/10

b/10  
b/10

### Objective:

- To get a grip on problem solving involving repetition structure.

### Devise Solution of the following problems using Pseudo Code

- Test/run the following pseudo code. If it contains any grammar/syntax error then mention that error otherwise give output on screen and variable status in RAM. Assume that user input oa = 624

1- Declare oa, r, da, exp  
 2- exp = 1  
 3- Print "Enter number :"  
 4- Input oa  
 5- da = 0  
 6- Repeat until oa >= 1  
 Y= 1 6.1- r = oa % 10  
 62 6.2- oa = oa / 10  
 4 6.3- da = da + r \* exp  
 6.4- exp = exp \* 8

7- exp = r  
 8- r = oa  
 9- oa = exp  
 10- Print "Number : ", da

~~624~~ 624

Console/Monitor/Screen

Enter number

| RAM   |
|---|
| $oa = 624, 62$<br>$r = 4$<br>$da = 0, 4$<br>$exp = 1$ |

- Input 'N' numbers from user and sum up only those numbers; which are odd. Also display the count of odd numbers entered.
- Input N numbers from user and find the largest value entered by user.
- Input a number N and display all the positive divisors of N.
- Calculate the factorial of a positive integer entered by user.

- The kingdom of ABRACADABRA, where the unit of currency is abra, has the following income tax code:  
 First 5,000 abra: 0% tax  
 Next 10,000 abra: 10% tax  
 Next 20,000 abra: 15% tax  
 Abras after 35,000: 20% tax  
 For example, someone earning 38,000 abra would owe: 10/2



$$5,000 * 0.00 + 10,000 * 0.10 + 20,000 * 0.15 + 3,000 * 0.20 = 4600 \text{ abras}$$

Write a pseudo code that ask user about his income and report the tax that he has to pay.

7. Input two numbers 'a' and 'n' from user. And calculate  $a^n$ .
8. Write a pseudo code to enter any number and calculate product of its digits.
9. Input a number and find whether the number is prime or not.
10. Assuming the ocean's level is currently rising at about 1.5 millimeters per year; write a pseudo code that displays a table showing the number of millimeters that the ocean will have risen each year for the next 25 years.

11. Input a number and display its equivalent octal number. ✓

12. Write a program, which takes a number as input from user and display the first five odd numbers smaller than the given number. Stop taking input when user enters five odd numbers less than the given number. X

13. Calculate the Grade Point Average (GPA) for a semester. GPA for a semester is calculated by:
  - a. Multiplying grade points with the credit hours in each course to obtain total grade points
  - b. Add up the total grade points to cumulative Grade Points and divide by the total number of credit hours in order to calculate the GPA for a semester.

For Example:

| Subject | Marks | Cr. Hrs. | Grade Point |
|---------|-------|----------|-------------|
| ITC     | 70    | 3        | 3           |
| PF      | 87    | 3        | 4           |
| EMT     | 83    | 4        | 3.7         |
| PF Lab  | 66    | 1        | 2.6         |

$(3*3+3*4+4*3.7+1*2.6)/11 =$   
 $39.4/11 = 3.49 \text{ GPA}$

Ask the user about the number of subjects he studied in the semester and then ask for the marks in these subjects and their credit hours in order to calculate GPA.

14. Write a program, which takes input of two integers. It then displays the greatest common divisor of the two entered integers.

15. The Last Stop Boutique is having a five-day sale. Each day, starting on Monday, the price will drop 10% of the previous day's price. For example, if the original price of a product is \$20.00, the sale price on Monday would be \$18.00 (10% less than the original price). On Tuesday the sale price would be \$16.20 (10% less than Monday). On Wednesday the sale price would be \$14.58; on Thursday the sale price would be \$13.12, and on Friday the sale price would be \$11.81. Develop a solution that will calculate the price of an item for each of the five days, given the original price.



*I never teach my pupils; I only attempt to provide the conditions in which they can learn.*

... ALBERT EINSTEIN ...

**Objective:**

- To get a grip on problem solving involving repetition structure.

**Devise Solution of the following problems using Pseudo Code**

1. How many beans are needed to arrange them in a triangular pattern. For example, for triangle of size/height 4, 10 beans are needed as shown in figure. You are asked to write a pseudo code which takes as input the size of a triangle and output the number of beans needed to arrange given beans in triangular pattern.



2. A perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself. For Example 6 is perfect number.  
Your task is to write a pseudo code, which input a number from user and tells whether it's a perfect number or not.

3. Write a program which takes input a number and tells whether its palindrome or not. Note that the number could be of any digit length.

4. Write a program that prompts the user to input an integer and then outputs both the individual digits of the number and the sum of the digits. For example, it should output the individual digits of 3456 as 3 4 5 6, output the individual digits of 8030 as 8 0 3 0, output the individual digits of 2345526 as 2 3 4 5 5 2 6, output the individual digits of 4000 as 4 0 0 0, and output the individual digits of -2345 as 2 3 4 5.

5. Write a program, which takes input of a number, and then display the equivalent binary number of it. For Example: if entered number is 11 then the answer will be 1101. *Ans*

6. Two integers 'a' and 'b' are said to be co-prime, relatively prime or mutually prime, if the only positive integer that evenly divides both of them is 1. Write a program which takes input from user for a and b and tells whether they are co-prime or not. For Example 14 and 15 are co-prime.

$$\begin{aligned} 35 &= 7 \times 5 \times 1 \\ 39 &= 13 \times 3 \times 1 \end{aligned}$$

7. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...  
By considering the terms in the Fibonacci sequence, find the sum of the first 'N' even-valued terms.

8. Write a program in which user keep entering the heights of people came for medical checkup. Program will stop taking input if user will enter -99. After this the program should display the smallest of all heights entered by the user, and the tallest of all heights entered by the user and also display the average height of the persons.

9. Write a program, which takes input of two integers. It then displays the least common multiple of the two entered integers.

10. A drunkard begins walking aimlessly, starting at a lamp post. At each time step, the drunkard forgets where he or she is, and takes one step at random, either north, east, south, or west. How far will the drunkard be from the lamp post after N steps?

Write a pseudo code, which first asks the number of steps that drunkard man has taken. It then asks for the steps taken by the drunkard on the





grid (Cartesian coordinate system).

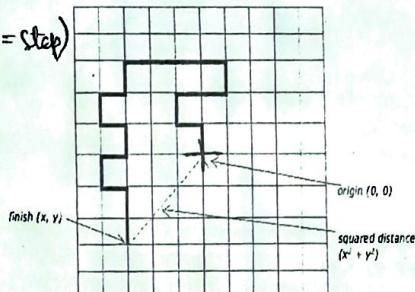
It is assumed that the lamp post position is at origin (0, 0). After inputting/taking all the moves from the drunkard, your pseudo code should print the square of the final distance from the origin.

To further understand, you may look at the sample run and the grid image.

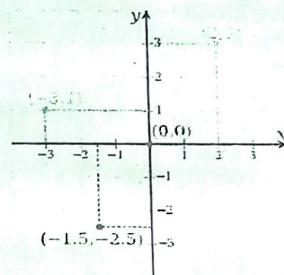
| Sample Run 1  | Sample Run 2  |
|---|---|
| Enter number of Steps: 20<br>0 1<br>-1 1<br>-1 2<br>0 2<br>1 2<br>1 3<br>0 3<br>-1 3<br>-2 3<br>-3 3<br>-3 2<br>-4 2<br>-4 1<br>-3 1<br>-3 0<br>-4 0<br>-4 -1<br>-3 -1<br>-3 -2<br>-3 -3<br><br>Squared distance = 18 | Enter number of Steps: 10<br>0 -1<br>0 0<br>0 1<br>0 2<br>-1 2<br>-2 2<br>-2 1<br>-1 1<br>-2 1<br>-3 1<br><br>Squared distance = 10 |

Steps

Grid Image for Sample Run 1



Cartesian coordinate system



(0,0)

And some pictures for your motivation.



The best angle from which to approach any problem is the try-angle.  
...Anonymous...





**Objective:**

- To get a grip on problem solving involving repetition structure.

**Devise Solution of the following problems using flow charts/Pseudo Code**

1. Write a pseudo code that asks the user to enter the amount that he or she has budgeted for a month. A loop should then prompt the user to enter each of his or her expenses for the month and keep a running total. When the loop finishes (decide a sentinel value yourself), the program should display the amount that the user is over or under budget.
2. Write a program that requests the user to enter two integers. The program should then calculate and report the sum of all the integers between and including the two integers. At this point, assume that the smaller integer is entered first. For example, if the user enters **2** and **9**, the program should report that the sum of all the integers from 2 through 9 is 44.
3. In a sumac sequence,  $t_1, t_2, \dots, t_m$ , each term is an integer greater than or equal 0. Also, each term, starting with the third, is the difference of the preceding two terms (that is,  $t_{n+2} = t_n - t_{n+1}$  for  $n \geq 1$ ). The sequence terminates at  $t_m$  if  $t_{m-1} < t_m$ . For example, if we have 120 and 71, then the sumac sequence generated is as follows:  
$$120, 71, 49, 22, 27.$$
This is a sumac sequence of length 5.
4. Write a program which calculates the sum of following series:  
$$1^2 + 2^3 + 3^4 + \dots + N^{N+1}$$
You will take value of 'N' from user.
5. Write a program which calculates the sum of first 'N' term of the following series:  
$$1^{S+1} + 2^{S+1} + 3^{S+2} + 2^{S+3} + 3^{S+5} + 2^{S+8} + 3^{S+13} + \dots$$
You will take value of 'N' and 'S' from user.
6. Write a program, which prints all the prime numbers in the range of two given numbers m and n.
7. Display the prime factors of a given positive integer.
8. Write a program which computes the following

$$4 * \sum_{k=1}^{10^6} \frac{(-1)^{k+1}}{2k-1} = 4 * \left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \dots\right)$$

9. Write a program that uses nested loops to collect data and calculate the average rainfall over a period of years. The program should first ask for the number of years. The outer loop will iterate once for each year. The inner loop will iterate twelve times, once for each month. Each iteration of the inner loop will ask the user for the inches of rainfall for that month. After all iterations, the program should display the number of months, the total inches of rainfall, and the average rainfall per month for the entire period.

***Input Validation:***

*Do not accept a number less than 1 for the number of years.*

*Do not accept negative numbers for the monthly rainfall*



**10.** Strong numbers are the numbers whose sum of factorial of digits is equal to the original number. Example: 145 is a strong number because  $1! + 4! + 5! = 145$ . Your task is to write a pseudo which checks whether a given number is strong number or not.

**11.** A dual prime is 2 prime numbers that are exactly "2" apart. Example: 3, 5 and 11, 13, etc. In this problem, you need to display all the dual primes up to a given number 'N'.



Success isn't a result of spontaneous combustion. You must set yourself on fire.

خدا تجھے کسی طوفان سے آشنا کر دے  
کہ تیرے بھلی بوجیں میں اخطراب نہیں  
تجھے کتابے ملکن نہیں شاید کہ تو  
کتاب خواں ہے مگر صاحب کتاب نہیں!



Dr. Allama Muhammad Iqbal (November 9, 1877 ~ April 21, 1938)

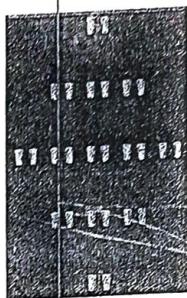
Objective:

- To get a review of print statement.
- To get a review of arithmetic operators and data types.
- To get an initial/quick review of taking input from keyboard.

1. Write a program that prints *Bjarne* (creator of C++) on the console in the following manner:

|         |       |       |       |       |        |
|---------|-------|-------|-------|-------|--------|
| BBBBBBB | JJJJJ | AAAAA | RRRRR | N N   | EEEEEE |
| B B     | J     | A A   | R R   | NN N  | E      |
| BBBBBBB | J     | AAAAA | RRRRR | N N N | EEE    |
| B B     | J J   | A A   | R R   | N N N | E      |
| BBBBBBB | JJJJ  | A A   | R R   | N NN  | EEEEEE |

2. Write a program that produces the following output:



3. A customer in a store is purchasing five items. The prices of the five items are:

Price of item 1 = \$12.95  
Price of item 2 = \$24.95  
Price of item 3 = \$6.95  
Price of item 4 = \$14.95  
Price of item 5 = \$3.95

Write a program that holds the prices of the five items in five variables. Display each items price, the subtotal of the sale, the amount of sales tax, and the total. Assume the sales tax is 6%.

4. Write code to declare variables as described below but give them meaningful names and data types.
- a. A variable to store the student quiz marks.
  - b. A variable to store the student test marks.
  - c. A variable to store the number of student in class.
  - d. A variable to store the number of miles traveled.
  - e. A variable to store the average speed of car.

5. Write a program that calculates the squares and cubes of the integers from 0 to 5 and uses tabs to print the following neatly formatted table of values:



| integer | square | cube |
|---------|--------|------|
| 0       | 0      | 0    |
| 1       | 1      | 1    |
| 2       | 4      | 8    |
| 3       | 9      | 27   |
| 4       | 16     | 64   |
| 5       | 25     | 125  |

6. Give meaningful names and appropriate data types for the following variables.
  - a. A variable to store the discounted price of an item.
  - b. A variable to store the number of juice bottles.
  - c. A variable to store the number of miles traveled.
  - d. A variable to store the highest test score.
7. Convert the following pseudo-code to C++ code. Be sure to declare the appropriate variables.  
Store 20 in the *speed* variable.  
Store 10 in the *time* variable.  
Multiply *speed* by *time* and store the result in the *distance* variable.  
Display the contents of the *distance* variable.
8. Convert the following pseudo-code to C++ code. Be sure to define the appropriate variables.  
Store 172.5 in the *force* variable.  
Store 27.5 in the *area* variable.  
Divide *area* by *force* and store the result in the *pressure* variable.  
Display the contents of the *pressure* variable.
9. To get the average of a series of values, you add the values up and then divide the sum by the number of values. Write a program that stores the following values in five different variables: 28, 32, 37, 24, and 33. The program should first calculate the sum of these five variables and store the result in a separate variable named *sum*. Then, the program should divide the *sum* variable by 5 to get the average. Display the average on the screen. ✓
10. Write a program that stores an five-digit integer in a variable, separates the integer into its individual digits and prints the digits separated from one another by three spaces each.
11. Suppose an employee gets paid every two weeks and earns \$1700.00 each pay period. In a year the employee gets paid 26 times. Write a program that declares the following variables:  
payAmount This variable will hold the amount of pay the employee earns each pay period. Initialize the variable with 1700.0.  
payPeriods This variable will hold the number of pay periods in a year. Initialize the variable with 26.  
annualPay This variable will hold the employee's total annual pay, which will be calculated.  
The program should calculate the employee's total annual pay by multiplying the employee's pay amount by the number of pay periods in a year, and store the result in the *annualPay* variable. Display the total annual pay on the screen.
12. One metric ton is approximately 2205 pounds. Write a program that creates a variable, which stores any amount of rice, in pounds, in a bag. The program outputs the number of bags needed to store one metric ton of rice.
13. One acre of land is equivalent to 43,560 square feet. Write a program that calculates the number of acres in a tract of land with 389,767 square feet. You are not allowed to declare any variable or constant.
14. Write a program that computes the tax and tip on a restaurant bill for a customer with a \$40 meal charge. The tax should be 6.75 percent of the meal cost. The tip should be 15 percent.

the total after adding the tax. Display the meal cost, tax amount, tip amount, and total bill on the screen.

15. Newton's law states that the force, F, between two bodies of masses M<sub>1</sub> and M<sub>2</sub> is given by:

$$F = k ((M_1 M_2) / d^2)$$

In which k is the gravitational constant and d is the distance between the bodies. The value of k is approximately  $6.67 \times 10^{-8}$  dyn. cm<sup>2</sup>/g<sup>2</sup>. Write a program that stores the masses of bodies in variables and the distance between the bodies. The program then output the force between the bodies.

16. Write a program that stores the radius of a circle in a variable and prints the circle's diameter, circumference and area. Use the constant value 3.14159 for 'pi'. Do all calculations in output statements.

17. A soft drink company recently surveyed 12,467 of its customers and found that approximately 14 percent of those surveyed purchase one or more energy drinks per week. Of those customers who purchase energy drinks, approximately 64 percent of them prefer citrus flavored energy drinks. Write a program that displays the following:

- o The approximate number of customers in the survey who purchase one or more energy drinks per week
- o The approximate number of customers in the survey who prefer citrus flavored energy drinks

18. Write a C++ program that stores the elapsed time for an event in a variable. The program then outputs the elapsed time in hours, minutes, and seconds. (For example, if the elapsed time is 9630 seconds, then the output is 2:40:30)

19. Assume that the following variables are defined:

```
int age;  
double pay;  
char section;
```

Write a single cin statement that will read input into each of these variables.





**Objective:**

- To get a review of taking input from keyboard, and basic concept of type coercion and type casting.
- Issues related to keyboard input stream.

**Task-1:**

Complete the following table by writing the value of each expression in the Value column according C++ language rules.

| Expression               | Value |
|--------------------------|-------|
| $28 / 4 - 2$             | 5     |
| $6 + 12 * 2 - 8$         | 28    |
| $4 + 8 * 2$              | 20    |
| $6 + 17 \% 3 - 2$        | 5     |
| $2 + 22 \% (9 - 7)$      | 48    |
| $(8 + 7) * 2$            | 30    |
| $(16 + 7) \% 2 - 1$      | 0     |
| $12 / (10 - 6)^2 / 9 =$  | 0     |
| $(19 - 3) * (2 + 2) / 4$ | 16    |
| $5 \% 10 \% 3$           | 2     |

**Task-3:**

Assume a program has the following variable definitions:

int units;  
float mass;

double weight;

and the following statement:

weight = mass \* units;

Which automatic data type conversion will take place?

- mass is demoted to an int, units remains an int, and the result of mass \* units is an int.
- units is promoted to a float, mass remains a float, and the result of mass \* units is a float.
- units is promoted to a float, mass remains a float, and the result of mass \* units is a double.

**Task-4:**

Assume a program has the following variable definitions:

int a, b = 2;

float c = 4.2;

and the following statement:

a = b \* c;

What value will be stored in a?

A. 8.4

B. 8

C. 0

D. None of the above

**Task-5:**

Assume that qty and salesReps are both integers. Use a type cast expression to rewrite the following

statement so it will no longer perform integer division.

unitsEach = qty / salesReps;

int

**Task-6:**

Each of the following programs has some errors. Locate as many as you can.

**Program-1**

```
using namespace std;
int void main ()
{
    double number1, number2, sum;
```

#include <iostream>



```
cout << "Enter a number: ";
>> cin << number1;
cout << "Enter another number: ";
>> cin << number2;
number1 + number2 = sum, Sum = number1 + number2
cout "The sum of the two numbers is " << sum();
return 0;
```

### Program-2

```
#include <iostream>
using namespace std;
int void main()
{
    int number1, number2;
    float quotient;
    cout << "Enter two numbers and I will divide\n";
    cout << "the first by the second for you.\n";
    cin >> number1, number2;
    //In your book following syntax is used
    //quotient = float<static_cast>(number1) / number2;
    //but in class initially we shall use C Style type cast
    quotient = (float)number1 / number2;
    cout << quotient
}
```

### Task-7: Miles per Gallon

Write a program that calculates a car's gas mileage. The program should ask the user to enter the number of gallons of gas the car can hold, and the number of miles it can be driven on a full tank. It should then display the number of miles that may be driven per gallon of gas.

### Task-8:

Suppose x and y are int variables and ch is a char variable. Consider the following input:  
5 28 36

What value (if any) is assigned to x, y, and ch after each of the following statements executes? (Use the same input for each statement).

- A. cin >> x >> y >> ch;
- B. cin >> ch >> x >> y;
- C. cin >> x >> ch >> y;
- D. cin >> x >> y;  
cin.get(ch);

### Task-9:

Suppose x and y are int variables and z is a double variable. Assume the following input data:  
37 86.56 32

What value (if any) is assigned to x, y, and z after each of the following statements executes? (Use the same input for each statement.)

- A. cin >> x >> y >> z;
- B. cin >> x >> z >> y;
- C. cin >> z >> x >> y;

### Task-10:

Suppose x and y are int variables and ch is a char variable. Assume the following input data:  
13 28 D

14 E 98

A B 56

What value (if any) is assigned to x, y, and ch after each of the following statements executes? (Use the same input for each statement.)



- A. 

```
cin >> x >> y;
cin.ignore(50, '\n');
cin >> ch;
```
- B. 

```
cin >> x;
cin.ignore(50, '\n');
cin >> y;
cin.ignore(50, '\n');
cin.get(ch);
```
- C. 

```
cin >> y;
cin.ignore(50, '\n');
cin >> x >> ch;
```
- D. 

```
cin.get(ch);
cin.ignore(50, '\n');
cin >> x;
cin.ignore(50, 'E');
cin >> y;
```

### Task-11:

Given the input:

46 A 49

and the C++ code:

```
int x = 10, y = 18;
char z = '*';
cin >> x >> y >> z;
cout << x << " " << y << " " << z << endl;
```

What is the output?

### Task-12:

Suppose that age is an int variable and name is a C style string variable char name[50]. What are the values of age and name after the following input statements execute?

```
cin >> age;
cin.getline(name, 50);
if the input is;
```

- A. 23 Lance Grant

OR the input is

- B. 23
   
Lance Grant

### Task-13:

During each summer, John and Jessica grow vegetables in their back yard and buy seeds and fertilizer from a local nursery. The nursery carries different types of vegetable fertilizers in various bag sizes. When buying a particular fertilizer, they want to know the price of the fertilizer per pound and the cost of fertilizing per square foot. The following program prompts the user to enter the size of the fertilizer bag, in pounds, the cost of the bag, and the area, in square feet, that can be covered by the bag. The program should output the desired result. However, the program contains logic errors. Find and correct the logic errors so that the program works properly.

```
#include <iostream>
#include <iomanip>
using namespace std;
void main()
{
    double cost;
```



```
double area;
double bagSize;
cout << fixed << showpoint << setprecision(2);
cout << "Enter the amount of fertilizer, in pounds, "<< "in one bag: ";
cin >> bagSize;
cout << endl;
cout << "Enter the cost of the " << bagSize << " pound fertilizer bag: ";
cin >> cost;
cout << endl;
cout << "Enter the area, in square feet, that can be "<< "fertilized by one bag: ";
cin >> area;
cout << endl;
cout << "The cost of the fertilizer per pound is: $"<< bagSize / cost << endl;
cout << "The cost of fertilizing per square foot is: $"<< area / cost << endl;
}
```

#### Task-14: Currency

Write a program that will convert U.S. dollar amounts to Japanese yen and to Euros, storing the conversion factors in the constants YEN\_PER\_DOLLAR and EUROS\_PER\_DOLLAR. To get the most up-to-date exchange rates, search the Internet using the term currency exchange rate. If you cannot find the most recent exchange rates, use the following:

1 Dollar = 83.14 Yen

1 Dollar = 0.7337 Euros

Format your currency amounts in fixed-point notation, with two decimal places of precision, and be sure the decimal point is always displayed.

#### Task-15: Monthly Payments

The monthly payment on a loan may be calculated by the following formula:

$$\text{Payment} = \frac{\text{Rate} * (1 + \text{Rate})^N}{((1 + \text{Rate})^N - 1)} * \text{L}$$

Rate is the monthly interest rate, which is the annual interest rate divided by 12. (12% annual interest would be 1 percent monthly interest.) N is the number of payments and L is the amount of the loan. Write a program that asks for these values and displays a report similar to:

|                        |             |
|------------------------|-------------|
| Loan Amount:           | \$ 10000.00 |
| Monthly Interest Rate: | 1%          |
| Number of Payments:    | 36          |
| Monthly Payment:       | \$ 332.14   |
| Amount Paid Back:      | \$ 11957.15 |
| Interest Paid:         | \$ 1957.15  |

*Wahi hai sahib-e-imroz (today's master) jis ne apni mehnat se  
Zamaane ke samundar se nikaala gohar-e-fardan (tomorrows pearl)  
[- DR ALLAMA MUHAMMAD IQBAL -]*

Objective:

- To get a grip on Repetition Structure: specifically, *while loop*.
- This practice includes both simple and nested loop.

Note: You are not allowed to use math library function without my permission.

#### Task-1

Do the Tasks given in Practice file 2 (13,14), Practice file 3 (5, 6, 9), Practice file 4 (especially: 5, 10), Lab 2 (especially 3, 5).

#### Task-2

Perform a statistical experiment by developing a solution to flip a coin (use `rand()` function) a given amount of times and then print the number of heads and the number of tails.

#### Task-3

Write a program, which prints the  $N^{\text{th}}$  terms of the Fibonacci sequence. If  $N = 5$  then your program display 3.

Fibonacci Sequence: 0, 1, 1, 2, 3, 5, 8, 13, 21, ...  
 1 2  
 2 3

#### Task-4

Using loops: do the task titled ABRACADABARA given in Practice-7.  
 When you done with this then change the problem by taking range from user instead of fixing it from 1 to 1000.

Fahim, bilal, zahid 5

#### Task-5

A teacher has asked all her students to line up according to their first name. For example, in one class Amy will be at the front of the line and Yolanda will be at the end. Write a program that prompts the user to enter the number of students in the class, and then loops to read in that many names. Once all the names have been read in, it reports which student would be at the front of the line and which one would be at the end of the line. You may assume that no two students have the same name.

*Input Validation: Do not accept a number less than 1 or greater than 25 for the number of students.*

Fahim, bilal, zahid 5

#### Task-6

Write a program, which calculates the sum of first 'N' term of the following series:

$$1^0 - 1^1 + 2^1 - 3^2 + 5^3 - 8^5 + 13^8 - 21^{13} + \dots$$

#### Task-7

Write a program that uses nested loops to collect data and calculate the average rainfall over a period of years. The program should first ask for the number of years. The outer loop will iterate once for each year. The inner loop will iterate twelve times, once for each month. Each iteration of the inner loop will ask the user for the inches of rainfall for that month. After all iterations, the program should display the number of months, the total inches of rainfall, and the average rainfall per month for the entire period.

*Input Validation:*

- Do not accept a number less than 1 for the number of years.
- Do not accept negative numbers for the monthly rainfall

#### Task-8

A long-distance carrier charges the following rates for telephone calls:

| Starting Time of Call | Rate per Minute |
|-----------------------|-----------------|
| 00:00-06:59           | 0.12            |
| 07:00-19:00           | 0.55            |
| 19:01-23:59           | 0.35            |



Write a program that asks for the starting time and the number of minutes of the call and displays the charges. The program should ask for the time to be entered as a floating-point number in the form HH.MM. For example, 07:00 hours will be entered as 07.00, and 16:28 hours will be entered as 16.28.

**Input Validation:** The program should not accept a time value, which is greater than 23:59. Also, no number whose last two digits are greater than 59 should be accepted.

*Note: Call can continue for more than one day*

### Task-9

Do circular prime question given in your quiz.

### Task-10

Write C++ programs to draw the following shapes.

The program should be generic: which means that your program will ask from the user about height (N) of pyramid/triangle/square and will display the shape accordingly.

#### Shape-1

For N=5

.....1  
....2,  
...3.  
.4..  
5....

#### Shape-2

For N=5

55555  
4444  
333  
22  
1

#### Shape-3

For N=5

----\*  
---\*\*  
---\*\*\*  
-\*\*\*\*  
\*\*\*\*\*

#### Shape-4

For N=5

1 2 3 4 5  
6 7 8 9  
10 11 12  
13 14  
15

#### Shape-13

For N=5

1  
1 1  
1 1 2  
1 1 2 2  
1 1 2 2 3

#### Shape-14

For N=5

1  
0 1  
1 0 1  
0 1 0 1  
1 0 1 0 1

#### Shape-15

For N=5

1 4 16 36 64  
1 9 25 49  
1 4 16  
1 9  
1

#### Shape-16

For N=5

1  
232  
34543  
4567651  
567898765

Shape-5  
For N=5

1\*2\*3\*4\*5  
1\*2\*3\*4  
1\*2\*3  
1\*2  
1

Shape-6

For N=5

1  
121  
12321  
1234321  
123454321  
1234321  
12321  
121  
1

Shape-7

For N=5

1 1  
2 1 1 2  
3 2 1 1 2 3  
4 3 2 1 1 2 3 4  
5 4 3 2 1 1 2 3 4 5

Shape-8

For N=5

123456789  
2345678  
34567  
456  
5

Shape-9

For N=5

123456789  
\_2345678  
\_\_34567  
\_\_\_456  
\_\_\_\_5

Shape-10

For N=5

5  
654  
76543  
8765432

Shape-17

For N=4

1\*\*\*\*\*1  
12\*\*\*21  
123\*321  
1234321

Shape-18

For N=5

\* \* \* \* \*  
\* \* \* \*  
\* \* \* \*  
\* \* \* \*  
\* \* \* \* \*

Shape-19

For N=6

\*  
\*\*  
\*  
\*\*\*  
\*  
\*\*\*\*

Shape-20

For N=8

\* , + +  
\*\* + +  
\*\*\* +  
\*\*\*\*  
+ \*\*\*  
++ \*\*  
+++ \*  
++++

Shape-21

For N=7

13579  
35791  
57913  
79135  
91357

Shape-22

For N=4

A C E G  
A C E  
A C  
A



987654321

### Shape-11

For N=6

100000  
010000  
001000  
000100  
000010  
000001

### Shape-12

For N=8

00000000  
00000001  
00000010  
00000011  
00000100  
00000101  
00000110  
00000111

### Shape-23

For N=7

ABCDEFGFEDCBA  
ABCDEF FEDCBA  
ABCDE EDCBA  
ABCD DCBA  
ABC CBA  
AB BA  
A A

### Shape-24

For N=7

\*  
\* \*  
\* \*  
\* \*  
\* \*  
\* \*  
\*

### Shape-25

For N=4

|    |    |    |    |    |
|----|----|----|----|----|
| 1  | 2  | 3  | 4  | 5  |
| 10 | 9  | 8  | 7  | 6  |
| 11 | 12 | 13 | 14 | 15 |
| 12 | 13 | 14 | 15 | 16 |
| 21 | 22 | 23 | 24 | 25 |

### Task-11

A number is Sparse if there are no two adjacent 1's in its binary representation. For example 5 (binary representation: 101) is sparse, but 6 (binary representation: 110) is not sparse.

Given a number x, find the smallest Sparse number which is greater than or equal to x.

Examples:

Input: x = 6

Output: Next Sparse Number is 8

Input: x = 4

Output: Next Sparse Number is 4

Input: x = 38

Output: Next Sparse Number is 40

Input: x = 44

Output: Next Sparse Number is 64

### Task-12

Given a number 'N', find the smallest number 'P' such that if we multiply all digits of 'P', we get 'N'. The result 'P' should have at least two digits.

Examples:

Input: n = 36

Output: p = 49

// Note that 4\*9 = 36 and 49 is the smallest such number

Input:  $n = 100$   
Output:  $p = 455$   
// Note that  $4^5 \cdot 5 = 100$  and 455 is the smallest such number

Input:  $n = 1$   
Output:  $p = 11$   
// Note that  $1 \cdot 1 = 1$

Input:  $n = 13$   
Output: Not Possible

### Task-13

Given an integer, find the most occurring digit in it. If two or more digits occur same number of times, then return the highest of them. Input integer is given as an int variable.  
Example:

Input:  $x = 12234$   
Output: The most frequent digit is 2

Input:  $x = 1223377$   
Output: The most frequent digit is 7

Input:  $x = 5$   
Output: The most frequent digit is 5

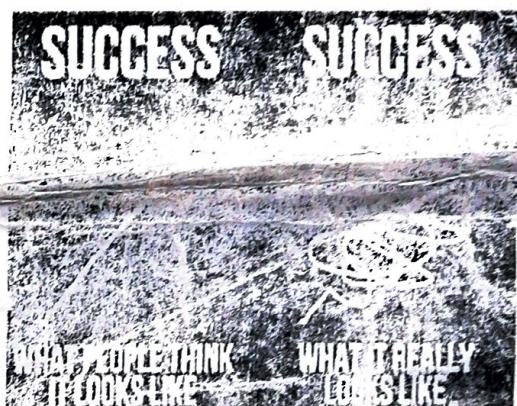
Input:  $x = 1000$   
Output: The most frequent digit is 0



Even the Ghulib was interested in Lamps ☺



ہزاروں خواہیں ایسی گئے  
ہر خواہش یہ دم نکلے  
بہت نکلے مرے ارمان لیکن پھر بھی کم نکلے



It's not that I am so smart; it's just that  
I stay with problems longer.

—Albert Einstein





### Objective:

- The objective of this lab is to get a grip on defining user define functions.
- Defining the prototypes appropriately.

Note: You are not allowed to use library function without my permission.

### Task-1:

Write a function, which display the multiplication table of a given number from 1 to 10.

void displayTable ( int )

### Task-2:

Write a function, which prints a diamond on the screen according to the height assed to it.

#### Sample Outputs:

If height passed = 9

4 5 6  
3 \* 7 1  
\*\*\* 3  
\*\*\*\*\* 5  
\*\*\*\*\* 7  
\*\*\*\*\* 9  
\*\*\*\*\* 7  
\*\*\*\* 5  
\*\*\* 3  
\* 1

X  
X XX  
X XX X X  
X X X X X X X  
X X X X X X X X  
X X X X X X X X X  
X X X X X X X X X X  
 $4 = 64 = 8^2$   
X X X X X X X X X X  
 $4 = 16 = 4^2$   
X X X X X X X X X X  
 $4 = 4 = 2^2$   
X X X X X X X X X X  
 $4 = 1 = 1^2$   
X X X X X X X X X X  
 $4 = 1 = 1^2$

### Task-3:

Write a function, which return true if the number passed to it is even otherwise return false.

bool isEven ( int )

$0 \rightarrow 2147483647$   
 $-2147483648$

### Task-4:

Write a function, which receives two integers and return the count of carry operation during their addition.

int countCarrys ( int , int )

Carefully decide the prototype of the following/remaining functions yourself and do verify it from Remember to make the Functions atomic too.

### Task-5:

Write a function, which checks: whether a given number is circular prime or not. It returns true if given number is circular prime otherwise return false.

$$C = \frac{5}{9} (F - 32)$$

$$f = \frac{9}{5} C + 32$$

### Task-6:

The formula for converting a temperature from Fahrenheit to Celsius is where F is the Fahrenheit temperature and C is the Celsius temperature. Write a function named convertToCelsius that accepts Fahrenheit temperature as an argument.

The function should return the temperature, converted to Celsius.

Demonstrate the function by calling it in a loop that displays a table of the Fahrenheit temperature through 20 and their Celsius equivalents.

### Task-7:

Write a program that asks the user to enter an items wholesale cost and its markup percentage. It should then display the items retail price. For example:

- If an item's wholesale cost is 5.00 and its markup percentage is 100%, then the item's retail price is 10.00.



- If an item's wholesale cost is 5.00 and its markup percentage is 50%, then the item's retail price is 7.50.

The program should have a function named calculateRetail that receives the wholesale cost and the markup percentage as arguments, and returns the retail price of the item.

Input Validation: Do not accept negative values for either the wholesale cost of the item or the markup percentage.

#### Task-8:

Write a function named as 'add', which receives two integers and returns the answer of addition of received numbers.

In Case, if the answer of addition overflow (does not fit in int) then your program should display a message "Integer Overflow" and exits the program using exit(0) function (which terminates your program immediately) rather than returning some value from function. It does the same for integer underflow but print the message "Integer Underflow".

Note: you are not allowed to use any data type other than int in your function.

#### Task-9:

Write a program to compute the cosine of x. The user should supply x and a positive integer n. We compute the cosine of x using the series and the computation should use all terms in the series up through the term involving x

$$\cos(x) = 1 - x^2/2! + x^4/4! - x^6/6! - \dots$$

#### Task-10:

A famous conjecture, called the Goldbach conjecture, says that every even integer greater than 2 has the property that it is the sum of two prime numbers. Computers have been used extensively to test this conjecture. No counterexample has been found. Write a program that will prove that the conjecture is true for all even integers in the given range.

It return 1 if you find the conjecture true otherwise it returns that even number which does not hold the above said conjecture.

#### Task-11:

The following formula gives the distance between two points,  $(x_1, y_1)$  and  $(x_2, y_2)$  in the Cartesian plane:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

iven the center and a point on the circle, you can use this formula to find the radius of the circle. /rite a program that prompts the user to enter the center and a point on the circle. The program should then output the circle's radius, diameter, circumference, and area. Your program must have at least the following functions:

21

2111

T1-2

- calculateDistance: This function takes as its parameters four numbers that represent two points in the plane and returns the distance between them.
- calculateRadius: This function takes as its parameters four numbers that represent the center and a point on the circle, calls the function distance to find the radius of the circle, and returns the circle's radius.
- calculateCircumference: This function takes as its parameter a number that represents the radius of the circle and returns the circle's circumference. (If r is the radius, the circumference is  $2 * \pi * r$ .)
- calculateArea: This function takes as its parameter a number that represents the radius of the circle and returns the circle's area. (If r is the radius, the area is  $\pi * r^2$ .)

sume that  $\pi = 3.1416$ .



### Task-12:

Write a function, which receives following three arguments from the user:

- Number
- Source base
- Target base

It then converts the number from source base to targeted base and returns the resultant number in target base.

Note: You can assume that the base will be less than or equal to 10. The default source\_base will be decimal and default target\_base will be binary.

**“It's hard enough to find an error in your code  
when you're looking for it; it's even harder  
when you've assumed your code is error-free.”**

--- Steve McConnell ---



### Objective:

- To get a grip on Array Structure manipulation (1-D Array).

### Task-1

Write a function, which finds the minimum value in an integer array.

### Task-2

Write a function, which finds the last occurrence of the minimum value in an integer array.

For Example:

If the array contains 10, 12, 9, 78, 90, 9, 78, 90

Then function should return 5, as 9 is the minimum value with its last occurrence at index 5.

### Task-3

Write a function, which finds whether a given array is sorted (ascending or descending both should be considered sorted) or not.

### Task-4

Write a function, which receives two arrays, and swap the contents of two equal size arrays.

### Task-5

Write a function which checks whether array elements are unique or not.

### Task-6: Poll Results

Forty students were asked to rate the quality of the food in the student cafeteria on a scale of 1 to 10 (1 meaning awful and 10 meaning excellent). Write a program which asks the 40 students to take part in poll and the summarize results of the poll as follows.

Here is one of the possible sample output after taking input/ratings from 40 students.

| Rating | Frequency |
|--------|-----------|
| 1      | 2         |
| 2      | 2         |
| 3      | 2         |
| 4      | 2         |
| 5      | 5         |
| 6      | 11        |
| 7      | 5         |
| 8      | 7         |
| 9      | 1         |
| 10     | 3         |

### Task-7:

Traditional password entry schemes are susceptible to "shoulder surfing" in which an attacker watches an unsuspecting user enter their password or PIN number and uses it later to gain access to the account. One way to combat this problem is with a randomized challenge-response system. In these systems, the user enters different information every time based on a secret in response to a randomly generated challenge. Consider the following scheme in which the password consists of a five-digit PIN number (00000 to 99999). Each digit is assigned a random number that is 1, 2, or 3. The user enters the random numbers that correspond to their PIN instead of their actual PIN numbers.



For example, consider an actual PIN number of 12345. To authenticate the user would be presented with a screen such as:

PIN: 0 1 2 3 4 5 6 7 8 9  
NUM: 3 2 3 1 1 3 2 2 1 3

The user would enter 23113 instead of 12345. This doesn't divulge the password even if an attacker intercepts the entry because 23113 could correspond to other PIN numbers, such as 69440 or 70439. The next time the user logs in, a different sequence of random numbers would be generated, such as:

PIN: 0 1 2 3 4 5 6 7 8 9  
NUM: 1 1 2 3 1 2 2 3 3 3

Your program should simulate the authentication process. Store an actual PIN number in your program. The program should use an array to assign random numbers to the digits from 0 to 9. Output the random digits to the screen, input the response from the user, and output whether or not the user's response correctly matches the PIN number.

**"To understand a program, you must become both the machine and the program."**

--- Alan Perlis ---



### Objective:

- Focus on parallel 1-D arrays and partially filled arrays.

### Task-1: *Lottery Application*

Write a program that simulates a lottery. The program should have an array of five integers named *lottery* and should generate a random number in the range of 0 through 9 for each element in the array. The user should enter five digits, which should be stored in an integer array named *user*. The program is to compare the corresponding elements in the two arrays and keep a count of the digits that match. For example, the following shows the lottery array and the user array with sample numbers stored in each. There are two matching digits (elements 2 and 4).

Lottery Array: 

|   |   |   |   |   |
|---|---|---|---|---|
| 7 | 4 | 9 | 1 | 3 |
|---|---|---|---|---|

 User Array: 

|   |   |   |   |   |
|---|---|---|---|---|
| 4 | 2 | 9 | 7 | 3 |
|---|---|---|---|---|

The program should display the random numbers stored in the lottery array and the number of matching digits. If all of the digits match, display a message proclaiming the user as a grand prize winner.

### Task -2

Write a program that uses the following arrays:

- *empId*: an array of seven long integers to hold employee identification numbers.  
The array should be initialized with the following numbers:  
5658845      4520125      7895122      8777541      8451277      1302850      7580489
- *hours*: an array of seven integers to hold the number of hours worked by each employee.
- *payRate*: an array of seven doubles to hold each employee's hourly pay rate
- *wages*: an array of seven doubles to hold each employee's gross wages

The program should relate the data in each array through the subscripts. For example, the number in element 0 of the 'hours' array should be the number of hours worked by the employee whose identification number is stored in element 0 of the 'empId' array. That same employee's pay rate should be stored in element 0 of the 'payRate' array.

The program should display each employee number and ask the user to enter that employee's hours worked and pay rate. It should then calculate the gross wages for that employee (hours times pay rate) and store them in the wages array. After the data has been entered for all the employees, the program should display each employee's identification number and gross wages.

*Input Validation: Do not accept negative values for hours or numbers less than 6.00 for pay rate.*

### Task -3: *Driver's License Exam*

The local Driver's License Office has asked you to write a program that grades the written portion of the driver's license exam. The exam has 20 multiple-choice questions.

Here are the correct answers:

1. B    6. A    11. B    16. C
2. D    7. B    12. C    17. C
3. A    8. A    13. D    18. B
4. A    9. C    14. A    19. D
5. C    10. D    15. D    20. A

Your program should store the correct answers shown above in an array. It should ask the user to enter the student's answers for each of the 20 questions, and the answers should be stored in another array. After the student's answers have been entered, the program should display a message indicating whether the student passed or failed the exam. (A student must correctly answer 15 of the 20 questions to pass the exam.) It should then display the total number of correctly answered questions, the total number of incorrectly answered questions, and a list showing the question numbers of the incorrectly answered questions.

*Note: Remember, we don't know what char arrays are so don't use them yet.*

**"Talk is cheap. Show me the code."**

--- Linus Torvalds ---



/\*  
Student Database

Code for the Case Study that we discussed on 9th-May-2018, Wednesday to understand parallel arrays, partially filled arrays and to understand the importance of separating interface from implementation (Presentation Layer vs. Business Layer)  
We also discussed how to attack a problem by breaking it into sub-problems.

```
/*
#include <iostream>
#include <iomanip>
using namespace std;

int addStudentRecord(int[], float [], int *, int, int, float);
int findStudentRollNo(int[], int, int);
bool isFull(int, int);
bool isEmpty(int);
bool removeStudentRecord(int[], float[], int *, int);
void displayAllRecords(int [], float [], int);

void studentDataBase();
int main()
{
    studentDataBase();
    return 0;
}
void studentDataBase()
{
    const int size=60;
    int rollNo[size];
    float cgpa[size];
    int recordCount = 0;

    int choice;
    int rno;
    float gp;
    bool appRunFlag = true;

    do
    {
        cout<<"\n\n||||||| Student Database |||||||||";
        cout<<"\nEnter 1 --> Add Record:";
        cout<<"\nEnter 2 --> Search Record:";
        cout<<"\nEnter 3 --> Display All Records:";
        cout<<"\nEnter 4 --> Display Records with CGPA >= X :";
        cout<<"\nEnter 5 --> Display Records with CGPA <= X :";
        cout<<"\nEnter 6 --> Display Class Average CGPA:";
        cout<<"\nEnter 7 --> Remove a Record with RollNo = X:";
        cout<<"\nEnter 8 --> Update CGPA with RollNo = X:";
        cout<<"\nEnter 0 --> Close the Application";
        cin>>choice;

        switch(choice)
        {
            case 1:
            {
                cout<<"\nEnter Roll No : ";
                cin>>rno;
                cout<<"\nEnter CGPA : ";
                cin>>gp;
            }
        }
    } while(appRunFlag);
}
```



```
int status = addStudentRecord(rollNo, cgpa, &recordCount, size, rno,
gp);
    if (status==1)
        cout<<"\n** Record Successfully Added **";
    else if (status==2)
        cout<<"\n** Database is Full **";
    else
        cout<<"\n** Record with Roll No: "<<rno<<" already exists **";
    break;
}
case 2:
{
    break;
}
case 3:
{
    displayAllRecords(rollNo, cgpa, recordCount);
    break;
}
case 4:
{
    break;
}
case 5:
{
    break;
}
case 6:
{
    break;
}
case 7:
{
    cout<<"\nEnter Roll No to Remove Record: ";
    cin>>rno;
    int status = removeStudentRecord(rollNo, cgpa, &recordCount, rno);
    if (status==true)
        cout<<"\n** Record Successfully Removed **";
    else
        cout<<"\n** Record not Found **";
    break;
}
case 8:
{
    break;
}
case 0:
{
    appRunFlag=false;
    break;
}
}
}
while(appRunFlag);
}
int addStudentRecord(int rollNo[], float cgpa[], int *rc, int size, int rno, float gp)
{
    if ( isFull(*rc, size) )
        return 2;
```



```
int recordIndex = findStudentRollNo(rollNo, *rc, rno);
if ( recordIndex != -1 ) //duplicate record
{
    return 0;
}

rollNo[*rc] = rno;
cgpa[*rc] = gp;
*rc = *rc + 1;
return 1;
}
int findStudentRollNo(int rollNo[], int rc, int key)
{
    int i = 0;
    while( i < rc && rollNo[i] != key)
    {
        i = i + 1;
    }
    if (i==rc)
        return -1;
    else
        return i;
}
bool isFull(int rc, int size)
{
    return rc == size;
}
void displayAllRecords(int rollNo[], float cgpa[], int rc)
{
    cout<<"\nRoll No          CGPA";
    cout<<"\n-----      -----";
    cout.fill('.');
    for ( int i=0; i < rc; i=i+1 )
    {
        cout<<"\n" <<setw(7)<<left<<rollNo[i]<<setw(12)<<right<<cgpa[i];
    }
}
bool removeStudentRecord(int rollNo[], float cgpa[], int *rc, int key)
{
    int index = findStudentRollNo(rollNo, *rc, key);
    if(index == -1)
        return false;
    /*
    I am using data shifting approach here. You may use other approach i.e. to move
    the last record on the index of the record to be deleted
    */
    for ( int i = index; i < *rc - 1; i = i + 1 )
    {
        rollNo[i] = rollNo[i+1];
        cgpa[i] = cgpa[i+1];
    }
    *rc = *rc - 1;
    return true;
}
```

### Complete all remaining functionalities

Inside every well-written large program is a well-written small program.

--- Charles Antony Richard Hoare ---



### Objective:

- To get a grip on Array Structure manipulation (2-D Array).

### Task - 1: 2-D Array Operations

Write a program that creates a two-dimensional array initialized with test data. Use any data type you wish (except `char`). The program should have the following functions:

- `getTotal`. This function should accept a two-dimensional array as its argument and return the total of all the values in the array.
- `getAverage`. This function should accept a two-dimensional array as its argument and return the average of all the values in the array.
- `getRowTotal`. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the total of the values in the specified row.
- `getColumnTotal`. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a column in the array. The function should return the total of the values in the specified column.
- `getHighestInRow`. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the highest value in the specified row of the array.
- `getLowestInRow`. This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the lowest value in the specified row of the array.

Demonstrate each of the functions in this program.

### Task - 2: Saddle Points

Write a program to search for the "saddle points" in a 5 by 5 array of integers. A saddle point is a cell whose value is greater than or equal to any in its row, and less than or equal to any in its column. There may be more than one saddle point in the array. Print out the coordinates of any saddle points your program finds. Print out "No saddle points" if there are none.

### Task - 3: Maze Runner

|    |    |    |    |    |
|----|----|----|----|----|
| 34 | 21 | 32 | 41 | 25 |
| 14 | 42 | 43 | 14 | 31 |
| 54 | 45 | 52 | 42 | 23 |
| 33 | 15 | 51 | 31 | 35 |
| 21 | 52 | 33 | 13 | 23 |

Do you like treasure hunts? In this Task you are to write a program to explore the above array for a treasure. The values in the array are clues. Each cell contains an integer between 11 and 55; for each

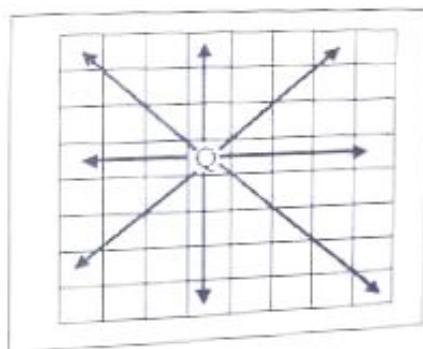


value the ten's digit represents the row number and the unit's digit represents the column number of the cell containing the next clue. Starting in the upper left corner (at 1,1), use the clues to guide your search of the array. (The first three clues are 11, 34, 42). The treasure is a cell whose value is the same as its coordinates. Your program must first read in the treasure map data into a 5 by 5 array. Your program should output the cells it visits during its search, and a message indicating where you found the treasure.

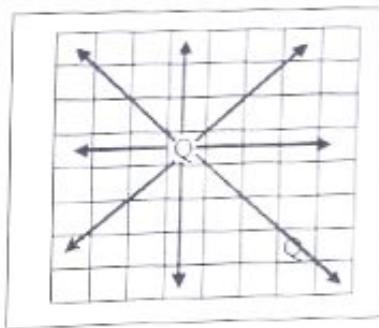
Note: The value in maze/2D-Array are not final, you may setup your own values in array, which obviously should not affect the logic to find the treasure.

#### Task - 4: Is the Queen safe?

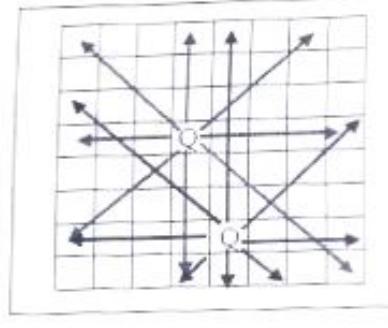
In the game of chess, a queen can attack pieces, which are on the same row, column, or diagonal. A chessboard can be represented by an 8 by 8 array. A 1 in the array represents a queen on the corresponding square, and a 0 in the array represents an unoccupied square. Your program is to read the location of two queens and then update the array appropriately. Then process the board and indicate whether it is safe to place the queen so that the other queen do not attack her.



Queen Movement



Queen not Safe



Queen Safe

#### Task - 5: Maximum Sum sub-matrix

Write a function, which receives 2-D array of size MxN and return maximum sum sub matrix.

For Example: in case of matrix-1, the program returns 15, and in matrix-2, the program returns 17.

|     |    |    |    |    |
|-----|----|----|----|----|
| 0   | -2 | -7 | 0  | -1 |
| 9   | 2  | -6 | 2  | 0  |
| -4  | 1  | -4 | 1  | 0  |
| -1  | 8  | 0  | -2 | 1  |
| -10 | 1  | 1  | -5 | 6  |
| -15 | -1 | 1  | 5  | -4 |

Matrix-1

|     |    |    |    |    |
|-----|----|----|----|----|
| 0   | -2 | -7 | 0  | -1 |
| 9   | 2  | -6 | 2  | 0  |
| -4  | 1  | -4 | 1  | 0  |
| -1  | 8  | 0  | -2 | 1  |
| -10 | 6  | 1  | -5 | 6  |
| -15 | -1 | 1  | 5  | -4 |

Matrix-2

"It's all talk until the code runs."

--- Ward Cunningham ---