



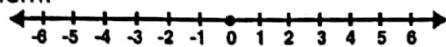
### Objective:

- To test/challenge your grip on problem solving by involving Sequence and Decision structure.

**Note →** If you feel any difficulties in implementing these tasks then feel free to discuss with me or with TAs but not with your peers at all.

### Challenge-1: Range Calculator (3)

Write a pseudo code which takes three integers named as a, b, and c on a number line and print the total integral numbers between them.



#### Assumption:

- $a < b < c$

#### Sample Run(s):

1	2
Please Enter the value of a: -4 Please Enter the value of b: -2 Please Enter the value of c: 5 Total numbers in the range are: 7	Please Enter the value of a: 1 Please Enter the value of b: 2 Please Enter the value of c: 4 Total numbers in the range are: 1

$$(b-a-1) + (c-b-1)$$

OR

$$c-a-2$$

### Challenge-2: Race Time (3)

Write a pseudo code that reports the time difference in seconds between the winner of a 5000m track race and one of the losing runners. It also report the relative increase in percent the difference represents.

The program's first ask for the input of winner's time (minutes, seconds, and hundredths of a second). The next three numbers in the input represents the loser's time (again in minutes, seconds, and hundredths of a second).

#### Sample Run(s):

1	2
Enter Winner's Time: 12 37 35 Enter Loser's Time: 13 8 77 Absolute Difference: 31.42 Relative Increase: 4.148%	Enter Winner's Time: 15 0 12 Enter Loser's Time: 18 59 0 Absolute Difference: 238.88 Relative Increase: 26.538%

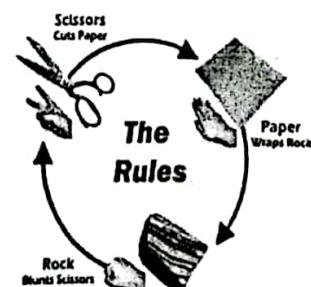
**Hint/Help:** How do I calculate a relative increase in percent? Divide the difference by the winning value and multiply by 100.

### Challenge-3: Rock Paper Scissor (6)

Write pseudo code to check the status of Rock-Paper-Scissor game. Your pseudo code will take input from two users about their choice (rock/paper/scissor) and will print the status of game.

#### Sample Run(s):

1	Enter First User Choice: Rock(R)/Paper(P)/Scissor(S): R Enter Second User Choice: Rock(R)/Paper(P)/Scissor(S): P Second Player Wins! Paper wraps Rock.
2	Enter First User Choice: Rock(R)/Paper(P)/Scissor(S): R Enter Second User Choice: Rock(R)/Paper(P)/Scissor(S): R Tie. Play again win the Game.



### Challenge-4: Gymnast's Score (6)

A gymnast's score is determined by a panel of 6 judges who each decide a score between 0 and 10. The final score is determined by discarding the high and low scores, and averaging the remaining 4. Write pseudo code that takes as input the scores of 6 judges for a gymnast and prints their average, after throwing out the high and low scores.

#### Sample Run(s):

1	Enter 1st Judge Score: 5 Enter 2nd Judge Score: 2 Enter 3rd Judge Score: 9 Enter 4th Judge Score: 7 Enter 5th Judge Score: 8 Enter 6th Judge Score: 7 Average Score after throwing out the highest and lowest scores is: 6.75
2	Enter 1st Judge Score: 7



Enter 2nd Judge Score: 7  
Enter 3rd Judge Score: 7  
Enter 4th Judge Score: 7  
Enter 5th Judge Score: 7  
Enter 6th Judge Score: 7  
Average Score after throwing out the highest and lowest scores is: 7

**Challenge-5: Count Carries**

(6)

Children are taught to add multi-digit numbers from right-to-left one digit at a time. Many find the "carry" operation - in which a 1 is carried from one digit position to be added to the next - to be a significant challenge. Your job is to count the number of carry operations for each of a set of addition problems so that educators may assess their difficulty.

For this purpose, you have to write a pseudo code, which takes two 3-digit numbers from user, and display the count of carry operations, which will be performed in adding the given numbers.

Sample Run(s):

1	Enter 1st Number: 123 Enter 2nd Number: 456 0 carry operations will be performed
2	Enter 1st Number: 555 Enter 2nd Number: 552 2 carry operations will be performed

**Resilience is very different than being numb.**

**Resilience means:**

**you experience, you feel, you fail, you hurt, You fall.**

**But, you keep going.**



### Objective:

- To test/challenge your grip on problem solving by involving Sequence, Decision and repetition structure.

*DeVise the solution of the following challenges using Pseudo code*

### Challenge-1! Candy Quest

(3)

You are a person who is always fond of eating candies. Your class fellow gave you a candy of length  $N$ , to eat during the break period of your school.

You start eating this candy from one of the ends. But as it is not your candy, your friend told you to eat exactly  $K$  unit length of candy during each bite. You will stop eating if the candy's length becomes 0. It means that you have eaten the entire candy. If at some point of time, the candy's length is positive, but less than  $K$ , you cannot take a bite thereafter.

Can you eat the complete candy? If yes, print the number of bites it will take, otherwise print "It is not possible to eat complete candy.".

Sample Run(s):

1	2
Enter Candy Length: 3	Enter Candy Length: 3
Enter Bite Length: 1	Enter Bite Length: 2
Minimum Bites Needed: 3	It is not possible to eat complete candy.

### Challenge-2! Even Digit Sum

(3)

Write pseudo code to input a number and prints the sum of even digits in the number.

Sample Run(s):

1	2
Please Enter a Number: 3492	Please Enter a Number: 379
Sum of Even Digits is: 6	Sum of Even Digits is: 0

### Challenge-3! Hockey Game Status

(4)

It's the hockey match finals in National Stadium Karachi and as always it has reached the penalty shootouts. Each team is given 5 shots to take and the team scoring a goal on the maximum number of shots wins the game. If both the teams' scores are equal, then the game is considered a draw and we would have 2 champions.

Write pseudo code which takes ten values from the user (Input value will be either 0 or 1, where 1 represents a goal and 0 represents a miss). First Input value will represent the status of penalty shot by 1<sup>st</sup> team and second Input value represents the penalty shot status by 2<sup>nd</sup> team, third Input value represents the shot status of 1<sup>st</sup> team and so on.

After taking ten values from user, your pseudo should report the winning team or match draw status if none is won.

Sample Run(s):

1	2
Enter penalty shot 1 status: 1	Enter penalty shot 1 status: 0
Enter penalty shot 2 status: 1	Enter penalty shot 2 status: 0
Enter penalty shot 3 status: 1	Enter penalty shot 3 status: 0
Enter penalty shot 4 status: 1	Enter penalty shot 4 status: 0
Enter penalty shot 5 status: 0	Enter penalty shot 5 status: 0
Enter penalty shot 6 status: 1	Enter penalty shot 6 status: 0
Enter penalty shot 7 status: 0	Enter penalty shot 7 status: 0
Enter penalty shot 8 status: 1	Enter penalty shot 8 status: 0
Enter penalty shot 9 status: 1	Enter penalty shot 9 status: 0
Enter penalty shot 10 status: 1	Enter penalty shot 10 status: 0
Game is won by second team,	Game is drawn,

### Challenge-4! Jerry, The Gambler!

(4)

Jerry has  $X$  dollars in debt to his bookie but currently has only  $Y$  dollars to his name. According to an astrologer, it's Jerry's lucky day today and he will win every game he steps in. A golden chance for Jerry to pay his debt  $\$0$ . Write a pseudo code which determines the minimum number of times you would have to go double-or-nothing at the roulette table (a gambling game) to pay your debt.

**Assumption:** Jerry goes "all-in" on every bet and wins every bet, and the casino allows bets of any value.



Sample Run(s):	
1	2
Enter Amount that Jerry has to its Name: 100 Enter Jerry's Debt Amount: 1000 After 1 bet(s), jerry balance: 200 After 2 bet(s), jerry balance: 400 After 3 bet(s), jerry balance: 800 After 4 bet(s), jerry balance: 1600 Number of Bet(s): 4	Enter Amount that Jerry has to its Name: 700 Enter Jerry's Debt Amount: 500 Number of Bet(s): 0



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

[ ... ALBERT EINSTEIN ... ]



### Objective:

- To test/challenge your grip on problem solving skills using repetition structure.

*Devise the solution of the following challenges using Pseudo code*

### Challenge-1: Marble Game

(5)

Write a pseudo code that allows two players to play the marble game. In the marble game, you ask the first player how many marbles to start with. Then, the game begins. The first player must take 1, 2 or 3 marbles. Then the second player goes and must take 1, 2 or 3 marbles. The winner is the player who takes the last marble. Allow two users to play this game and print out the winner (player #1 or player #2). Assume that both players enter valid inputs (1, 2 or 3, and they never try to take more marbles than there are in the pile.)

#### Sample Run

```
How many marbles will you be playing with? 10
Player # 1, there are 10 marbles left.
How many marbles will you take? 3
Player # 2, there are 7 marbles left.
How many marbles will you take? 2
Player # 1, there are 5 marbles left.
How many marbles will you take? 1
Player # 2, there are 4 marbles left.
How many marbles will you take? 1
Player # 1, there are 3 marbles left.
How many marbles will you take? 3
Player #1, you took the last marble and have won!
```

### Challenge-2: Chocolate Factory

(6)

Consider the following scenario:

Charlie once visited chocolate factory to buy some chocolates. Chocolate cost Rs. 3 each and he had only Rs. 45. He was disappointed. He wanted more. But when he reached the factory he was amazed. There was a scheme on that special day. If you return 3 wrappers of the chocolate you will get one chocolate for free. So how many chocolates can he buy?

#### Solution:

Charlie has Rs. 45 and each chocolate cost Rs. 3. So he buys only 15. But there is a scheme. He will return 15 wrappers and get 5 chocolates free. Then he will return 3 out of 5 and get one free. And he will again use one wrapper with the remaining 2 to get one more. So  $15+5+1+1 = 22$ .

#### Your Task:

Write a program, which ask user about the amount of rupees he has and cost per chocolate. In return your program will output the maximum amount of chocolates that can be bought considering the special offer discussed in above scenario.

#### Sample Run(s)

1	2
Enter Amount of Rupees: 45 Enter the cost per Chocolate: 3 Maximum Chocolates that can be bought are: 22	Enter Amount of Rupees: 100 Enter the cost per Chocolate: 13 Maximum Chocolates that can be bought are: 10

### Challenge-3: Mirroring Number

(7)

The number 89 is the first Integer with more than one digit that fulfills a special property which is as follows:

We get the same number if expressed in the form:  $= 8^1 + 9^2 = 89$

The next number having this property is 135, See this property again:  $135 = 1^1 + 3^2 + 5^3$

Write pseudo code which test whether a given number has such property or not.

#### Sample Run(s)

1	2
Enter Number: 175 It's a Mirroring Number	Enter Number: 123 It's not a Mirroring Number

### Challenge-4: Bit Status

(10) 7

Your job is to list all of the numbers up to  $2^N - 1$  that contains a 1 on given bit number in their binary representation of it.



**For Example:** If N=4 and bit number = 2, then your pseudo code will print [2, 3, 6, 7, 10, 11, 14, 15].  
The binary numbers from 1 to 16 are:

4 3 2 1 (bit #)

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

**Sample Run(s)**

1	2
Enter Value of N: 4 Enter the Bit # : 2 2, 3, 6, 7, 10, 11, 14, 15,	Enter Value of N: 3 Enter the Bit # : 1 1, 3, 5, 7

**"SOME OF THE BEST PROGRAMMING IS DONE ON PAPER, REALLY.  
PUTTING IT INTO THE COMPUTER IS JUST A MINOR DETAIL."**  
[... MAX KANAT-ALEXANDER, CODE SIMPLICITY: THE FUNDAMENTALS OF SOFTWARE ...]



### Objective:

- To test/challenge your grip on problem solving skills related to formatted IO and selection/decision structure In C++.

Devise the solution of the following challenges using C++ Language

### Challenge - 1:

(3)

Write a C++ program to compute a water and sewer bill. The input is the number of gallons consumed. A water and sewer bill is computed as follows:

- o The water charge is .021 dollars per 100 gallons
- o The sewer charge is .001 dollars per 100 gallons
- o A service charge of 2 percent is applied to the total of the water and sewer charges

Note: To get full credit, your program input/output should look like as per sample run. This instruction must be followed for any future tasks as well till the end of semester.

#### Sample Run

```
Enter number of gallons of water consumed: 100000
Water charge : Rs***21.00
Sewer charge : Rs***1.00
Service charge: Rs***0.44
Total charge : Rs***22.44
```

### Challenge - 2: Naughty Friend

(5)

Max has a rectangular prism with dimensions A, B, and C. He writes A, B, and C on a piece of paper. When he is not looking, Andrew replaces these numbers with  $X=A+B$ ,  $Y=B+C$ , and  $Z=A+C$ . Now, Max sees three numbers X, Y, Z on the piece of paper. But Max needs help computing the surface area of this rectangular prism. Help him!

Help him by writing a program in which Max will feed/store the values of X, Y, and Z and in return your program will display on console the values of A, B, C and also display the surface area of the prism.

Formula to calculate Surface Area =  $2(WL + HL + HW)$

Where W=width, H= height, and L=length

### Challenge - 3: Izzy Wizzy

(3)

We conducted a survey on newspaper subscriptions. More specifically, we asked each of the N respondents the following two questions:

- Question 1: Are you subscribing to Newspaper X?
- Question 2: Are you subscribing to Newspaper Y?

As the result, A respondents answered 'Yes' to Question 1, and B respondents answered 'yes' to Question 2. What are the maximum possible number and the minimum possible number of

respondents subscribing to both newspapers X and Y? Write a C++ program to answer this question.

Examples: (minimum from A & B) (If A>B then result otherwise)

1. If  $N=10$ ,  $A=3$ ,  $B=5$ , then maximum possible people = 3, minimum possible people = 0. zero
2. If  $N=10$ ,  $A=7$ ,  $B=5$ , then maximum possible people = 5, minimum possible people = 2.

### Challenge - 4: Task Scheduling

(4)

You have three tasks, all of which need to be completed.

First, you can complete any one task at cost 0.

Then, just after completing the  $i^{th}$  task, you can complete the  $j^{th}$  task at cost  $|A_j - A_i|$ .

Here,  $|x|$  denotes the absolute value of  $x$ .

Your C++ program will take three costs and will find and print the minimum total cost required to complete all the task.

Examples:

1. If  $A_1 = 1, A_2 = 6, A_3 = 3$  Then program outputs 5.
2. If  $A_1 = 11, A_2 = 5, A_3 = 5$  Then program outputs 6.

Minimum  
and Maximum



### Challenge - 5: Got any Grapes? ☺

(5)

For simplicity, we'll assume that there are only three types of grapes: green grapes, purple grapes and black grapes.

Fahad, Kashif and Hamza are all grapes' lovers; however, their preferences of grapes are different. To make all of them happy, the following should happen:

Fahad, Kashif and Hamza should eat at least  $X$ ,  $Y$  and  $Z$  grapes, respectively.

- Fahad has an extreme affinity for green grapes, thus he will eat green grapes and green grapes only.
- On the other hand, Kashif is not a fan of black grapes. Any types of grapes except black would do for him. In other words, Kashif can eat green and purple grapes.
- Hamza has a common taste. He enjoys grapes in general and will be pleased with any types of grapes, as long as the quantity is sufficient.

Knowing that his friends are so fond of grapes, Usman decided to host a grape party with them. He has prepared a box with  $A$  green grapes,  $B$  purple grapes and  $C$  black grapes.

However, Usman isn't sure if the box he prepared contains enough grapes to make everyone happy. Can you please find out whether it's possible to distribute grapes so that everyone is happy or Usman has to buy some more grapes?

It is not required to distribute all the grapes, so it's possible that some of them will remain unused. Usman will give 6 inputs to your program i.e.  $X$ ,  $Y$ ,  $Z$  and  $A$ ,  $B$ ,  $C$ . Print "Yes", if the grapes distribution is possible to make them happy otherwise print "No".

Examples:

- If  $X=1$ ,  $Y=6$ ,  $Z=2$  and  $A=4$ ,  $B=3$ ,  $C=3$ , It prints Yes.
- If  $X=5$ ,  $Y=1$ ,  $Z=1$  and  $A=4$ ,  $B=3$ ,  $C=2$ , It prints No.

# "First solve the problem. And then write the code."

[... JOHN JONSON ...]



### Objective:

- The objective of this lab is to test your logic and grammar knowledge related to repetition structure and basic use of user define functions.

### Challenge - 1: My CMath

(08)

Define your own functions as listed below. The listed function should behave like the function provided in C++.

**Note:** that both positive and negative values may be passed to these functions.

**Assumption:** values passed to these functions will be in the range of INT\_MIN and INT\_MAX.

- double truncateNumber(double);  
Computes the largest integer value not greater than the given argument.
- double ceilNumber(double);  
Computes the smallest integer that is greater than or equal to the given argument.
- double floorNumber(double);  
Computes the largest integer that is smaller than or equal to the given argument.
- double roundNumber(double);  
Computes the nearest integer value to the given argument, rounding halfway cases away from zero.

Sample Executions/Run(s)

Function Required	Output	Equivalent C++ Function	Output
1 cout << truncateNumber(25.83);	25	cout << trunc(25.83);	25
2 cout << truncateNumber(-2.9);	-2	cout << trunc(-2.9);	-2
3 cout << ceilNumber(2.001);	3	cout << ceil(2.001);	3
4 cout << floorNumber(2.001);	2	cout << floor(2.001);	2
5 cout << roundNumber(7.9);	8	cout << round(7.9);	8
6 cout << roundNumber(7.1);	7	cout << round(7.1);	7
7 cout << roundNumber(-7.5);	-8	cout << round(7.9);	-8
8 cout << roundNumber(-7.2);	-7	cout << round(-7.2);	-7

### Challenge - 2: Truncatable Prime

(06)

In number theory, a **left-truncatable prime** is a prime number which, in a given base, contains no 0, and if the leading ("left") digit is successively removed, then all resulting numbers are prime. For example, 9137, since 9137, 137, 37 and 7 are all prime.

A **right-truncatable prime** is a prime which remains prime when the last ("right") digit is successively removed. 7393 is an example of a right-truncatable prime, since 7393, 739, 73, and 7 are all prime.

A **left-and-right-truncatable** (or simply **truncatable prime**) prime is a prime which remains prime if the leading ("left") and last ("right") digits are simultaneously successively removed down to a one- or two-digit prime. 1825711 is an example of a left-and-right-truncatable prime, since 1825711, 82571, 257, and 5 are all prime.

**NOTE:** In base 10, there are exactly 4260 left-truncatable primes, 83 right-truncatable primes, and 920720315 left-and-right-truncatable primes. [[https://en.wikipedia.org/wiki/Truncatable\\_prime](https://en.wikipedia.org/wiki/Truncatable_prime)]

Write a function (prototype given below) which receives a base-10 number and reports that whether the received number is truncatable prime or not.

```
bool isTruncatable ( long long int );
```

### Challenge - 3: ABRACADABRA

(08)

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

Examples

Input	Output
1 36	49
2 100	455
3 1	11
4 13	Not Possible



#### Challenge - 4: Bumpy Number

(08)

Working from left-to-right if no digit is exceeded by the digit to its left it is called an **increasing number**; for example, 134468. Similarly if no digit is exceeded by the digit to its right it is called a **decreasing number**; for example, 66420. We shall call a positive integer that is neither increasing nor decreasing a "bumpy" number; for example, 155349.

Clearly there cannot be any bumpy numbers below one-hundred, but just over half of the numbers below one-thousand (525) are bumpy. In fact, the least number for which the proportion of bumpy numbers first reaches 50% is 538.

Surprisingly, bumpy numbers become more and more common and by the time we reach 21780 the proportion of bumpy numbers is equal to 90%.

Write code which finds the least number for which the proportion of bumpy numbers is exactly 99%.

You might not think that programmers are artists, but  
programming is an extremely creative profession.  
It's logic-based creativity.

[... JOHN ROMERO ...]



### Objective:

- Distributing the functionality keeping in view the atomicity, Presentation Layer and Business Layer and using by reference mechanism wherever needed.

### Challenge - 1: Digital Watch

(12.5(0.5, 4, 2, 3, 3))

In this challenge, you will develop a series of functions which demonstrate some basic operations of a digital watch.

1	<code>void printTimeIn24HoursFormat (int hours, int minutes, int seconds);</code>	Prints the received time on console in 24-hour format.
2	<code>void printTimeIn12HoursFormat (int hours, int minutes, int seconds);</code>	Prints the received time on console in 12-hour format.
3	<code>void incrementHours (int * hours, int * minutes, int * seconds, int increment = 1);</code>	Increment the value of hours as provided in increment parameter.
4	<code>void incrementMinutes (int * hours, int * minutes, int * seconds, int increment = 1);</code>	Increment the value of minutes as provided in increment parameter.
5	<code>void incrementSeconds (int * hours, int * minutes, int * seconds, int increment = 1);</code>	Increment the value of seconds as provided in increment parameter.

**Note:** To get full credit, develop a smart logic (make the machine do less work) to achieve the tasks listed in this challenge and closely observe and follow the sample run. Functions related to time increment does nothing if *incremental value* <= 0.

Sample Run

Sample Code	Console Output
<code>int hr1 = 4, min1 = 5, sec1 = 50;</code>	
<code>printTimeIn24HoursFormat (hr1, min1, sec1);</code>	04:05:50
<code>printTimeIn12HoursFormat (hr1, min1, sec1);</code>	04:05:50 AM
<code>int hr2 = 15, min2 = 2, sec2 = 45;</code>	
<code>printTimeIn12HoursFormat (hr2, min2, sec2);</code>	03:02:45 PM
<code>incrementHours (&amp;hr2, &amp;min2, &amp;sec2, 4);</code>	
<code>printTimeIn24HoursFormat (hr2, min2, sec2);</code>	19:02:45 PM
<code>incrementMinutes (&amp;hr1, &amp;min1, &amp;sec1, 100);</code>	
<code>printTimeIn12HoursFormat (hr1, min1, sec1);</code>	05:45:50 AM

### Challenge - 2: Change K<sup>th</sup> Digit

(5)

Write a C++ program which inputs three integers *n*, *k*, and *d*, where *n* is possibly a negative *int*, *k* is a non-negative *int*, and *d* is a non-negative single digit (between 0 and 9 inclusive), and it changes the contents of identifier *n* such that the *k*<sup>th</sup> digit is replaced with *d*. Counting starts at 0 and goes from right to left. It prints the updated value of *n* at the end. If value of *k* is invalid (think yourself that what is invalid for *k*) then function will simply return without doing anything.

Examples

- If *n* = 468, *k* = 0, *d* = 1 then the number becomes 461.
- If *n* = 468, *k* = 1, *d* = 1 then the number becomes 418.

The prototype of the function that will achieve this task is as follows:

```
void changeKthDigit(int * number, int kthDigit, int newDigit);
```

In order to achieve atomicity for the function *changeKthDigit*, following helper/utility function also needs to be complete.

```
int getNumberLength (int);
int getLargestNDigitNumber (int);
```

You have to decide yourself that how the provided utility functions will be helpful for the *changeKthDigit*.

Sample Run

Sample Code	Console Output
<code>int a = 123456;</code>	
<code>changeKthDigit(&amp;a, 1, 9);</code>	
<code>cout &lt;&lt; a &lt;&lt; '\n';</code>	123496
<code>cout &lt;&lt; getNumberLength(123456) &lt;&lt; " : " &lt;&lt; getLargestNDigitNumber(6);</code>	6 : 999999

**“Any fool can write code that a computer can understand.**

**Good programmers write code that humans can understand.”**

[... MARTIN FOWLER ...]



### Objective:

- we shall be looking at problems related to one-dimensional Array structure manipulation.

To get full credit in each of the challenges, your solution should be atomic, should also follow rest of conventions as discussed so far in lectures and labs.

### Challenge - 1: Missing Integer (4)

Write a function which receives an array of type int and returns the smallest missing positive integral value in it.

#### Examples

1   If array = {1, 2, 0}	3 is returned.
2   If array = {3, 4, -1, 1}	2 is returned.
3   If array = {7, 8, 9, 11, 12}	1 is returned.

### Challenge - 2: Friend Request (6)

You have been recently hired by Facebook and your first task is to analyze possible impact of its new expected policy regarding sending friend requests.

The new expected policy to be employed is as follows.

A person  $a$  cannot send a friend request to person  $b$  if any of the following condition is true:

1. age of  $b \leq (0.5 * \text{age of } a + 7)$
2. age of  $b > \text{age of } a$
3. (age of  $b > 100$ ) and (age of  $a < 100$ )

Facebook will provide you an array of integers whose each location represents the age of their user. So, if the array name is ages, then ages[i] represents the age of user  $i$  where  $0 \leq i \leq N$ .

#### Note:

- A person can't send request to itself.
- If  $a$  sends request to  $b$  then it is not necessary for  $b$  to send request to  $a$ .
- A person age will always be in the range  $1 \leq \text{age} \leq 120$ .

So, your job is to find the total number of friend requests possible under the new policy. Your function results will help upper management to decide something about implementing the new policy.

For this you are given following function which receives ages array and its size and returns the required information.

```
int getFriendRequestsCount (int ages [], int N);
```

#### Examples

1   If ages array = {20, 20}	2 is returned.
2   If ages array = {20, 30, 40}	1 is returned.

### Challenge - 3: Container Capacity (10)

Given  $N$  non-negative integers  $a_1, a_2, \dots, a_n$ , where each represents a point at coordinate  $(i, a_i)$ . Total  $N$  vertical lines are drawn such that the two endpoints of line  $i$  is at  $(i, 0)$  and  $(i, a_i)$ . Find two lines, which together with  $x$ -axis forms a container, such that the container contains the most water.

**Note:** You may not slant the container and  $N$  is at least 2.



The vertical lines are represented by array  $\{1, 8, 6, 2, 5, 4, 8, 3, 7\}$ . In this case, the max area of water (grey shaded section) the container can contain is 49.

To achieve above task, you need to implement following function.

```
int getMaxWaterArea(int arr[], int N);
```

**Magic happens when you don't give up, even though you want to. The universe always falls in love with a stubborn heart.**

[... ...]



## **Objectives:**

- Looking at an application of 1D Arrays: IEEE-754 Single Precision Conversion.

## IEEE-751 Conversion

Write a series of function as listed below: Their main goal is to implements the IEEE-754 single precision procedure as discussed in class.

1. float getFloatingPointValueIEEE754(bool []); (6)
    - o returns the floating-point value represented by received array.
    - o Received array size is 32.
    - o Index 0 of array represents the bit number 31 i.e., most significant bit (msb) and index 31 represents the bit number 0 i.e., least significant bit (lsb).
  2. void convertFloatingPointValueTo32BitRepresentationIEEE754(float, bool []); (12)
    - o populate the received array with 32-bit representation of received floating point value as per IEEE-754.
  3. void print32BitRepresentationIEEE754(bool []); (2)
    - o prints the received array.

Notes

- o Don't use any short cut to achieve the above tasks.
  - o You are required to implement procedure as discussed in class.
  - o You may define additional functions in order to make your work atomic and manageable.
  - o To get full marks, your console output should be similar to the sample run.

## Sample Run

```
1 int main()
2 {
3     bool arr[32] =
4         {0, 1, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
5     print32BitRepresentationIEEE754(arr);
6     float val = getFloatingPointValueIEEE754(arr);
7     cout << "\nFloating Point Value = " << val;
8     bool res[32];
9     convertFloatingPointValueTo32BitRepresentationIEEE754(12.375, res);
10    print32BitRepresentationIEEE754(res);
11 }
```

## Console Output

	SB	Exponent	Mantissa
4	0	1000 0010	100 0110 0000 0000 0000 0000
6	Floating Point Value = 12.375		
	SB	Exponent	Mantissa
9	0	1000 0010	100 0110 0000 0000 0000 0000

Just a funny story about Steve Jobs and Floating Point Numbers

"Steve Jobs calls one of the mysteries of life how his friend and Apple's Co-founder Steve Wozniak never built a floating point BASIC for the Apple II, despite Jobs having begged him for several weeks to do it. As a result of Wozniak's refusal, Job reached out to Microsoft to license Bill Gates' BASIC."

[https://en.wikipedia.org/wiki/Applesoft\\_BASIC](https://en.wikipedia.org/wiki/Applesoft_BASIC)

**"There are no traffic jams along the extra mile."**

By ROGER STAUBACH

**VERY FEW PEOPLE GO THE EXTRA MILE IN THEIR WORK. MAKE YOURSELF STAND OUT FROM THE CROWD BY DOING A BIT MORE THAN IS EXPECTED FROM YOU. THIS WILL ALWAYS WORK OUT TO YOUR BENEFIT.**



### Objective:

- Looking at problems related to 2-dimensional arrays.

### Challenge - 1: Most Loveable Food

(5)

A restaurant wants to know their most loved food/menu-item in their menu. For this, they surveyed from  $N$  customers about their  $M$  dishes in the menu. Your job is to analyze the survey and compute the most loveable food mentioned in it.

Restaurant will provide you a matrix of  $N$  by  $M$  where  $N$  represents the number of customers and  $M$  represents the dishes. If a dish is liked a customer, then the matrix entry will be true otherwise false.

**Example:** In the sample matrix below: there are 5 customers and 8 dishes and you can see that dish number 2 is most loveable dish.

	0	1	2	3	4	5	6	7
0	true	false	true	false	true	false	false	false
1	false	false	true	true	false	false	true	false
2	false	false	false	true	true	false	false	false
3	true							
4	false	false	true	false	false	false	false	true

Function Prototype: int getMostLoveableDish(bool mat[][], int N, int M);

### Challenge - 2: Rotate Matrix

(5)

Given an  $N$  order matrix, clockwise rotate elements in it.

Function Prototype: void rotateMatrix(int mat[][], int N);

Examples/Explanation	
For 3*3 matrix	Matrix Elements After Rotation:
1 2 3 4 5 6 7 8 9	4 1 2 7 5 3 8 9 6
For 4*4 matrix	Matrix Elements After Rotation:
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	5 1 2 3 9 10 6 4 13 11 7 8 14 15 16 12

سوداگری نہیں یہ عبادت خدا کی ہے  
اے بے خبر جزا کی تنا بھی چھوڑ دے  
واعظ ثبوت لانے جو مے کے جواز میں  
اقبال کو یہ ضد ہے کہ پینا بھی چھوڑ دے

[ڈاکٹر علام محمد اقبال]



### Objective:

- The objective of this lab is to explore an application of 1D/2D char Arrays.

### Challenge 1: SubStrings (6, 4, 3)

Provide definitions of the following three function as described.

1.1

```
int findSubString( const char * destination, const char * key, int start=0 );
key
A substring to search for.
destination
destination is the array in which searching has to be performed
start
The index of the character in the string to begin the search with, or 0 to
start from the beginning
return Value
The first character index of the string that matches the sub string, -1
otherwise
```

1.2

```
void printAllOccurrences( const char * destination, const char * key);
It prints all the occurrences of the given key in destination array.
key
A substring to search for.
destination
destination is the array in which searching has to be performed
```

#### Sample Run:

Sample Code	Console Output
cout << findSubString("BSEF22 Batch Morning Batch", "Batch") << '\n';	7
cout << findSubString("BSEF22 Batch Morning Batch", "tch") << '\n';	9
cout << findSubString("BSEF22 Batch Morning Batch", "Batch", 10) << '\n';	21
printAllOccurrences("HeHello how areheHehe HEyou", "He");	0, 2, 17,

1.3

Consider the 2D array (list/array of strings) declared and initialized in the sample run given below. You have to provide definition for the function named 'searchNamesList', which finds the given key in the list of names. It returns the row number in which the key is found otherwise returns -1.

**Note:** to get full credit, try to use what you have already implemented in this lab.

```
int searchNamesList(const char namesList[][20], int rows, const char * key);
```

#### Sample Run:

Sample Code	Console Output
int main()	
{	
char names[5][20] = {"hello Ok", "abcdef", "Ok", "Hello how are you",	
"The End"};	
cout << searchNamesList(names, 5, "Ok") << '\n';	0
cout << searchNamesList(names, 5, "are") << '\n';	3
return 0;	
}	

بھلیاں جس میں ہو آسودہ' وہ خرمن تم ہو  
بیچ کھاتے ہیں جو اسلاف کے مدفن' تم ہو

مونکو نام جو قبروں کی تجارت کر کے  
کیا نہ بیچو کے جو مل جائیں صنم پتھر کے

[ناکثر علامہ محمد اقبال]



### Objective:

- Exploring the 1D + 2D arrays on heap and struct usage.

### Challenge-1: Rotate Matrix

Write a C++ program, which accepts a 2D array of order R\*C and rotate the 2D elements by +90 degree. (10)

Note Following:

- The number of rows and columns of 2D array may be different. Hint: transpose
- You are not allowed to change the received matrix elements. So, you need to create a new 2D array (on heap obviously) to store the rotated elements and have to return it from function.

### For Example:

If the received matrix is:

	0	1	2
0	1	2	3
1	4	5	6
2	7	8	9

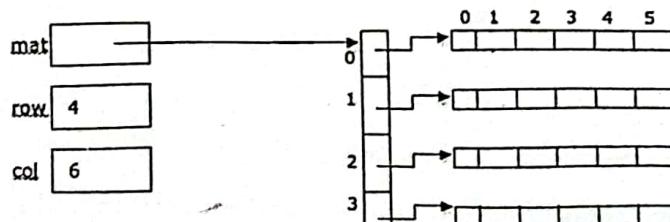
then it should be changed to following

	0	1	2
0	7	4	1
1	8	5	2
2	9	6	3

The function prototype is as follows:

```
int ** getRotatedMatrix ( const int * const * const mat, const int rows, const int columns, int
& newRows, int & newColumns );
```

Where first parameter 'mat' receives the 2D array. You may assume that the 2D that you receive is made through array of pointers as shown in the diagram below (also discussed in class). And 2<sup>nd</sup> and 3<sup>rd</sup> parameter represents the number of rows and columns in matrix respectively.



### Challenge-2: Rotate Matrix

(10)

Do the above challenge using Matrix struct as discussed in class.

```
Matrix getRotatedMatrix ( Matrix );
```

