



Objective:

- This quiz will check your status of learning so far in the course by asking questions from easy to difficult based on knowledge learned so far.

Challenge-A:

((9): 2,2,3,2)

Part-A

Give worst **time** and **space** bound of the following code segment?

```
void displayString(char s[])
{
    Stack<char> st;
    int i=0;
    while (s[i]!='\0')
    {
        st.push(s[i]);
        i++;
    }
    i=0;
    while(s[i]!='\0')
    {
        while(!st.isEmpty())
            cout<<st.pop();
        i++;
    }
}
```

Part-B

Consider a 3-D array of size $A \times B \times C$ with each dimension's index starts from k (can be positive or negative).

Devise a Row-Major mapping for $A \times B \times C$ array to 1-D array of size $A \times B \times C$ with 1-D array starting index = 0. Where index set is $[i][j][k]$.

Part-C

Devise a mapping for the N order symmetric sparse matrix in which non-zero elements are laid out as follows.

*				*
*			*	*
*		*		*
*	*			*
*				*

Sparse Matrix of Order 5: where * shows the non-zero values

Part-D

Consider the following Non-Symmetric Sparse Matrix of char values.

	0	1	2	3	4	5	6	7
0	a		b	c	b			x
1		f	r	k	o	p		
2		y						
3	s				t			
4			y				z	g
5	h			t	n			m
6								
7		e						

Fill the 1-D arrays below according to Compressed Sparse Row Format that we discussed in class.

Value

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Column_Index

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Row_Pointer

--	--	--	--	--	--	--	--

Challenge-C:

(6)

Write an efficient algorithm which receives a sorted array 'A' and a key 'k' and returns the index of the first occurrence of 'k' in 'A'. Return -1 if k does not appear in 'A'. For example: if received array is = {5, 7, 10, 20, 20, 40, 50, 60, 60, 60} and key = 20 then it returns 3.