Object-oriented Programming

Summer 2022
Assignment 1

Due Date: 25 July 2022

Total Marks:(60), Weightage: (6)

Paraphrasing Application

Paraphrasing is the process of rewording a text, often done for simplification or clarity. Implement a simple C++ console-based application for paraphrasing purposes, in the following manner.

1. Read a file that contains a list of words and synonyms, where each synonym is separated by a single white space character, as in the following example:

```
abandon discontinue vacate
absent missing unavailable
cable wire
calculate compute determine measure
safety security refuge
```

Consider use of parallel arrays as well as (dynamically allocated) array of arrays for storing the synonyms, once the file is read. For instance, the above file may be represented in the memory in the following manner

words array

0	abandon
1	absent
2	cable
3	calculate
4	safety

synonyms array of array

0	ptr0	→	discontinue	vacate	
1	ptr1	→	missing	unavailable	
2	ptr2	→	wire		
3	ptr3	→	compute	determine	measure
4	ptr4	→	security	refuge	

Where, ptr0, ptr1, etc. are pointers assigned at run-time.

2. Take a text as input from the user and tokenize it into words.
3. For each word (after step 2) that exists in the list of synonyms loaded (in step-1), replace it with corresponding synonym (selected randomly in case of multiple options)
4. Produce the output as paraphrased text

Task#2: Create a '**Set**' ADT, which will implement the basic operations of Set concept in mathematics. The data members and operations needed for this ADT are given below.

Data Members:

- int * data
- int noOfElements
- int capacity

Supported Operations:

The class '**Set**' should support following operations

1. **Set (int cap = 0);**
   Default parameterized constructor. Sets 'cap' to 'capacity' and initializes rest of the data members accordingly. If user sends any invalid value then sets the cap to default value.

2. **Set(const Set & ref);**
   overloaded copy constructor to implement deep copy.

3. **~Set()**
   Free the dynamically allocated memory.

4. **void insert (int element);**
   Stores the element in the Set.
   Remember! Set is a collection of distinct elements. If the array is full, then resize the capacity of Set by double.

5. **void remove (int element);**
   remove the element from the Set. If the noOfElements in the Set are equal to or less than 50% of capacity then shrink the Set by 25% i.e., if the capacity was 12 and noOfElements are less than equals to 6 then the new Set should be of the capacity equals to 9.

6. **int getCardinality() const;**
   returns the number of elements in the set.

7. **Set calcUnion (const Set & s2) const;**
   returns a new Set object which contains the union of 's2' set and calling object set.

8. **Set calcSymmetricDifference (const Set & s2);**
   returns a new Set object which contains the symmetric Difference of 's2' set and calling object set. Where symmetric difference is: $A \Delta B = A \cup B - A \cap B$

9. **Set calcDifference (const Set & s2) const;**
   returns a new Set object which contains the intersection of 's2' set and calling object set.

10. **int isMember (int val) const;**
    returns 1 if 'val' is member of the set otherwise return 0.

11. **int isSubSet (const Set & s2) const;**
    returns 1 if s2 is proper subset of calling object set, return 2 if improper subset otherwise returns 0.

12. **void reSize (int newcapacity);**
    resize the set to new capacity. Make sure that elements in old set should be preserved in the new set if possible.

13. **void update(int prVal, int curVal);**
    This function should replace the previous value(prVal) with current value (curVal). Make sure the property of set should not be violated (**distinct elements**). If the prVal is not present in the array, then display a message "**Target value not found**". In-case of successful modification then display a message "**Record update successfully**". If prVal is present in the set but curVal is also already available, then display a message "Violation set property. Can't Modify the data".

14. **void show() const;**
    This function should display the contents of Set for calling object.

Write main function to test the functionalities of supporting functions of Set ADT