PIC-10A: Homework 7

Due 11/27/21 11:59pm via CCLE

Problem 1:

Write a function `unique(A, size_A, size_unique)` that stores all unique elements **(remove duplicates)** of a given array of integers in a dynamic array, and return a pointer to this array.

```
int* unique(const int A[], int size_A, int &size_unique)
```

where `size_A` is the size of the integer array `A`, and `size_unique` is the size of the unique array.

Instruction & hint:

- Of course you do not know at the outset how many unique elements in A. The worst case is A contains all distinguished elements. Hence you can start by creating a dynamic array of the same size as `A`.
- Store all unique elements of `A` in this dynamic array. Record the number of unique elements in `size_unique`.
- Create another dynamic array where the size is exactly `size_unique`. Copy all elements of the first dynamic array to this new one.
- Remember to delete all temporary arrays.

Example: if A contains elements {1,2,3,4,3,5,2,1}, then the function returns a pointer to an int dynamic array that contains distinct elements {1,2,3,4,5}, and `size_unique = 5`.

Download the file *hw7_1.cpp*, write your function in this file. Save and submit your file as *hw7_1.cpp.*

Problem 2

Given definition of class Monster, implement its member functions

```
class Monster {
private:
    int pos_x=0;     // position of monster (x,y)
    int pos_y=0;     // initially set to (0,0)
    string name;     // monster name

public:
    // accessors
    string getName() const;
    int getX() const;
    int getY() const;
    void display() const;

    // mutators
    void setName(string aName);
    void setXY(int x, int y);
    void setX(int x);
    void setY(int y);
    void moveLeft();
    void moveRight();
    void moveUp();
    void moveDown();
};
```

The class describes monster's position with member functions as follows:

- `string getName() const`: return name of monster
- `int getX() const`: return x-position of monster
- `int getY() const`: return y-position of monster
- `void display() const`: display monster's name and position. For example, monster with name "dragon" and position (10,12), your function should print out:

  `Monster: dragon, position = (10,12)`
- `void setName(string aName)`: set name of monster to `aName`
- `void setXY(int x, int y)`: set position `pos_x` and `pos_y` of monster to `x` and `y` respectively.
- `void setX(int x)`: set `pos_x` to `x`
- `void setY(int y)`: set `pos_y` to `y`
- `void moveLeft()` : decrease `pos_x` by 1
- `void moveDown()` : decrease `pos_y` by 1
- `void moveRight()`: increase `pos_x` by 1
- `void moveUp()` : increase `pos_y` by 1

Note: positions of monsters need to be between 0 and 100 inclusively in both x and y directions. You must check whether the parameters in the `setX(int x)`, `setY(int y)` and `setXY(int x, int y)` functions satisfy this condition and whether a move is valid. If a move or position set is invalid, print out an error message.

- For example, function call `setX(-10)` is not eligible. Your function should not let this happen but print out an error message.
- Another example: if `pos_x=0`, then the function call `moveLeft()` does not allow the move but prints out an error message as well.

Download the file *hw7_2.cpp*, implement your class in this file. Save and submit your file as *hw7_2.cpp.*