CSCI 3333 Practice Quiz HASH2

For the problems here, use the following Record and HashTable class declarations.

```cpp
class Record {
    public:
        long int id;
        string name;
        string transcript;
};

class HashTable {
    public:
        HashTable();
        void insert(Record* r);
        Record* search(long int id);
    
    private:
        float load_factor();
        void rehash();

        Record** A; // For linear probing problems
        list<Record*>* A; // For chaining problems
        int capacity;
};
```

**Problem 1.** Complete the following implementation of the load_factor method using linear probing.

```cpp
float HashTable :: load_factor()
{
    int count = 0;

    for (int i = 0; i < _____; ++i)
        if (A[i] _____ nullptr)
            count = count _____ _____;

    return static_cast<float>(count) / this->____;
}
```
Problem 2. Determine the truth of the following statements about the running times of linear probing hash tables with load factor $\alpha$.

If $\alpha > 1/2$, then a collision has occurred. □ True □ False

The worst-case running time of $\text{insert}$ is $\Theta(\alpha)$. □ True □ False

The best-case running time of $\text{search}$ is $\Theta(1/(1 - \alpha))$. □ True □ False

The worst-case running time of $\text{search}$ is $\Theta(1/(1 - \alpha))$. □ True □ False

Problem 3. Complete the following implementation of the $\text{load factor}$ method using chaining.

```cpp
float HashTable :: load_factor()
{
    int count = 0;
    for (int i = 0; i < _____; ++i)
    {
        count = count + _____;_____;
    }
    return static_cast<float>(count) / this->_____;
}
```
Problem 4. Determine the truth of the following statements about chaining hash tables with load factor $\alpha$.

- It is true that $\alpha > 1$ □ always □ never □ sometimes
- For the worst-possible hash function, the worst-case running time of erase is □ $\Theta(1)$ □ $\Theta(\alpha)$ □ $\Theta(n)$
- For the worst-possible hash function, the best-case running time of insert is □ $\Theta(1)$ □ $\Theta(\alpha)$ □ $\Theta(n)$
- For the best-possible hash function, the worst-case running time of search is □ $\Theta(1)$ □ $\Theta(\alpha)$ □ $\Theta(n)$

Problem 5. Complete the following implementation of the rehash method using linear probing.

```cpp
void HashTable :: rehash()
{
    vector<Record*> V;
    for (int i = 0; i < _____; ++i)
        if (A[i] != nullptr)
            V.push_back(______);

    delete[] A;
    capacity *= 2;
    A = new _____[capacity];

    for (Record* r : _____)
        insert(r);
}
```