CS2134 Homework 2
Spring 2015
Due 11:59 p.m. Thurs February 12, 2015

February 8, 2015

Assignment 2 includes a programming portion and a written portion. The programming portion should consist of a single file (hw02.cpp), and the written portion should consist of a single file (hw02written) in a standard format (.txt, .doc, .htm., or .pdf). Be sure to include your name at the beginning of each file! You must hand in both files via NYU Classes.

Programming Part:

1. **Enter data from the file** `MTA_train_stop_data.txt`**.** The data from this assignment is from http://www.mta.info/developers/download.html. (Please note that we will only be using some of the information in this file for this assignment.[1])
   In the batch phase you will read all the data from the file called `MTA_train_stop_data.txt` into a container of type `vector<trainStopData>`.
   Your program will define the class `trainStopData`. It has the following private member variables :

```
string stop_id;    // id of train stop (1st token)
string stop_name;  // name of station (4th token)
double stop_lat;   // latitude of train stop location
double stop_lon;   // longitude of train stop location
```

   Your class should also have a constructor and the following public member functions:

```
string get_id( ) const
string get_stop_name( ) const
double get_latitude( ) const
double get_longitude( ) const
```

[1]The data is in a common format; please read https://developers.google.com/transit/gtfs/reference?csw=1 for more information.

Written Part

1. For each of the following code fragments, determine the worst case running time using **Big-Oh** notation as a function of n.

(a)
```
int sum = 0;
for (int k = n; k>=1; k = (int) k/3)
       sum += k;
cout << sum;
```

(b)
```
for (int j = 1; j <= n; j++)
{
        for (int i = j; i > 1; i/=9)
               cout << "(" << i << ", " << j << ") " ;
        cout << endl;
}
```

(c)
```
template <class Comparable>
int min(const vector<Comparable> & items)
{
    if (items.size() == 0) return -1;

    int min_index = 0;

    for (int i = 0; i < items.size(); ++i)
        if (items[i] < items[min_index])
            min_index = i;

    return min_index;
}
```

(d)
```
for( i = 0; i < n; ++i)
   for (j = 0; j < n; ++j)
      a[i][j] = b[i][j] + c[i][j];
```

(e)
```
void in_order( vector<int> & a )
{
     int j = 0;
     for( int p = 1; p < a.size( ); p++ )
     {
          print(a); // This function takes O(n) time.
          Comparable tmp = a[ p ];
          for( j = p; j > 0 && tmp < a[ j - 1 ]; j-- )
                    a[ j ] = a[ j - 1 ];
          a[ j ] = tmp;
     }
 }
```

2. What is the difference between `delete []` and `delete`?

3. For the following class, implement a deep `operator=`. Implement the method outside the class interface.

```
template< class Object >
class memoryBuffer{
   public:

    memoryBuffer(int size=10):theSize{size},objects{new Object[size]}{}
   ...   // other methods not omitted to save space
    memoryBuffer& operator=( const memoryBuffer& rhs );

  private:
   int theSize;
   Object* objects;
}

//write the implementation for the method here.
```