

Bangladesh University of Business and Technology
Department of Computer Science and Engineering
CSE 232: Data Structures Lab
Solution to Lab 05 Tasks

Task 1

Sort a list of strings lexicographically using -

- Bubble Sort
- Selection Sort
- Insertion Sort

Sample Input:	Sample Output:
7	
Luna	Draco
Draco	Ginny
Ginny	Harry
Tom	Lily
Ron	Luna
Harry	Ron
Lily	Tom

Using Bubble Sort:

```
1 #include <iostream>
2 using namespace std;
3
4 void BubbleSort(string s[], int n)
5 {
6     for (int i = 0; i < n - 1; i++)
7     {
8         bool swapped = false;
9         for (int j = 0; j < n - i - 1; j++)
10         {
11             if (s[j] > s[j + 1])
12             {
13                 string temp = s[j];
14                 s[j] = s[j + 1];
15                 s[j + 1] = temp;
16                 swapped = true;
17             }
18         }
19         if (swapped == false)
20             break;
21     }
22 }
23
24 int main()
25 {
26     int n;
27     cin >> n;
28     string str[n];
29     for (int i = 0; i < n; i++)
30         cin >> str[i];
31     BubbleSort(str, n);
32     for (int i = 0; i < n; i++)
33         cout << str[i] << endl;
34 }
```

Using Selection Sort:

```
1 #include <iostream>
2 using namespace std;
3 void SelectionSort(string s[], int n)
4 {
5     int i;
6     for (i = 0; i < n - 1; i++)
7     {
8         int min_idx = i;
9         for (int j = i + 1; j < n; j++)
10         {
11             if (s[j] < s[min_idx])
12                 min_idx = j;
13         }
14         if (min_idx != i)
15         {
16             string temp = s[min_idx];
17             s[min_idx] = s[i];
18             s[i] = temp;
19         }
20     }
21 }
22 int main()
23 {
24     int n;
25     cin >> n;
26     string str[n];
27     for (int i = 0; i < n; i++)
28         cin >> str[i];
29     SelectionSort(str, n);
30     for (int i = 0; i < n; i++)
31         cout << str[i] << endl;
32 }
```

Using Insertion Sort:

```
1 #include <iostream>
2 using namespace std;
3 void InsertionSort(string s[], int n)
4 {
5     for (int i = 1; i < n; i++)
6     {
7         string key = s[i];
8         int j = i - 1;
9         while (j >= 0 && s[j] > key)
10         {
11             s[j + 1] = s[j];
12             j--;
13         }
14         s[j + 1] = key;
15     }
16 }
17 int main()
18 {
19     int n;
20     cin >> n;
21     string str[n];
22     for (int i = 0; i < n; i++)
23         cin >> str[i];
24     InsertionSort(str, n);
25     for (int i = 0; i < n; i++)
26         cout << str[i] << endl;
27 }
```

Task 2

Suppose there are **n** coins on a table. Find out the *minimum* number of coins you need to take from those **n** coins so that the amount of money you have is *strictly greater* than the amount left on the table. The input consists of **n**. Followed by the monetary value of each of those coins. You should print the number of coins you need to take.

Sample Input: 2 3 3	Sample Output: 2
Sample Input: 3 2 1 2	Sample Output: 2
Sample Input: 20 7 84 100 10 31 35 41 2 63 44 57 4 63 11 23 49 98 71 16 90	Sample Output: 6

Sample Solution:

```
1 #include <iostream>
2 using namespace std;
3
4 void InsertionSort(int arr[], int n)
5 {
6     for (int i = 1; i < n; i++)
7     {
8         int key = arr[i];
9         int j = i - 1;
10        while (j >= 0 && arr[j] < key)
11        {
12            arr[j + 1] = arr[j];
13            j--;
14        }
15        arr[j + 1] = key;
16    }
17 }
18
19 int main()
20 {
21     int n, i, j, arr[100], cnt = 0, total = 0, me = 0;
22     cin >> n;
23     for (i = 0; i < n; i++)
24     {
25         cin >> arr[i];
26         total = total + arr[i];
27     }
28     total = total / 2;
29     InsertionSort(arr, n);
30     for (i = 0; i < n; i++)
31     {
32         me = me + arr[i];
33         cnt++;
34         if (me > total)
35             break;
36     }
37     cout << cnt << endl;
38 }
```