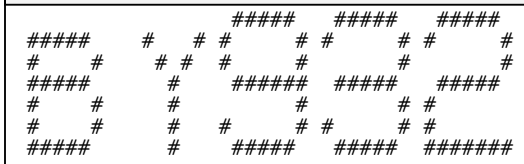

Oct 25 11:22 2014 hash.cpp Page 1

```
  1     /*******************************
  2     *Filename:hash.cpp              *
  3     *Login:by932                   *
  4     *AssignmentNo:ass4             *
  5     *DateLastModified:19/10/2014    *
  6     *******************************/
  7     #include <iostream>
  8     #include <string.h>
  9     #include <stdlib.h>
 10     #include "hash.h"
 11     using namespace std;
 12
 13     HT::HT()
 14     {
 15             hashtable[SIZE] = NULL;
 16
 17             for (int index = 0; index < SIZE; index++)
 18                     hashtable[index] = new htnode();
 19     }
 20
 21     HT::~HT()
 22     {
 23             for (int i =0; i < 100; i++)
 24                     delete  hashtable[i];
 25
 26     }
 27     void HT::calculate(char input[])
 28     {
 29             int sum = 0;
 30             int which = 0;
 31             for (int i = 0; i < 20; i++)
 32             {
 33                     sum += static_cast<int>(input[i]);
 34             }
 35
 36             which = sum % SIZE;
 37             hashtable[which]->data.load(input);
 38
 39     }
 40
 41     void HT::findtop()
 42     {
 43             for (int i = 0; i < SIZE; i++)
 44             {
 45                     topten(hashtable[i]->data);
 46             }
 47             sort(sortarr);
 48     }
 49     bool HT::topten(LIST& pass)
 50     {
 51             NPtr head;
 52             //head = hashtable->data.returnhead();
 53             if (head == NULL)
 54             {
 55                     return false;
 56             }
```

Oct 25 11:22 2014 hash.cpp Page 2

```
 57             NPtr temp = head;
 58             int move = 0;                           // do not change the head pointer
 59             while (temp != NULL)
 60             {
 61
 62                     while (sortarr[move].content != '\0' && move < 10)
 63                     {
 64                             move++;
 65                     }
 66                     if (sortarr[move].content == '\0')
 67                     {
 68                             strcpy(sortarr[move].content, temp->content);
 69                             sortarr[move].count = temp->count;
 70                             move++;
 71                     }
 72                     else
 73                     {
 74                             for (int i = 0; i < 10; i++)
 75                             {
 76                                     if (sortarr[i].count < temp->count)
 77                                     {
 78                                             strcpy(sortarr[i].content, temp->content);
 79                                             sortarr[i].count = temp->count;
 80                                     }
 81                             }
 82                     }
 83                     temp = temp->next;
 84
 85             }
 86             return true;
 87
 88     }
 89
 90     void HT::sort(sortnode right[])
 91     {
 92             for (int i = 9; i > 0; i--)
 93             {
 94                     for (int q = 0; q < 10; q++)// print out sorted collection
 95                     {
 96                         cout << right[q].count << " ";
 97                     }
 98                     cout << endl;
 99
100                     for (int j = 0; j < i; j++)             // perform sorting
101                     {
102                             if (right[j].count > right[j+1].count)
103                             {
104                                     int temp = right[j+1].count;
105                                     char tempch[21] = {'\0'};
106                                     strcpy(tempch, right[j + 1].content);
107
108                                     right[j+1].count = right[j].count;
109                                     strcpy(right[j + 1].content, right[j].content);
110
111                                     right[j].count = temp;
112                                     strcpy(right[j].content, tempch);
```

Oct 25 11:22 2014 hash.cpp Page 3

```
113                                 }
114
115                         }
116                 }
117         }
```

Oct 25 11:22 2014 list.cpp Page 1

```
 1     /*******************************
 2     *Filename:list.cpp              *
 3     *Login:by932                   *
 4     *AssignmentNo:ass4             *
 5     *DateLastModified:19/10/2014    *
 6     *******************************/
 7     #include <iostream>
 8     #include <cstring>
 9     #include "list.h"
10     using namespace std;
11
12     LIST::LIST()
13     {
14             nonodes = 0;
15             head = NULL;
16
17     }
18
19     LIST::~LIST()
20     {
21             node* temp = head;
22             while(temp && temp->next) {
23                     temp=temp->next;
24                     delete temp;
25             }
26
27             nonodes = 0;
28     }
29
30     bool LIST::load(char input[])
31     {
32             if (head == NULL)       // linked list is empty - so this will be the head node
33             {
34
35                     NPtr newnode = new node;
36
37                     if (newnode == NULL)            // could not allocate memory
38                     {
39                             cout << "Allocation error occured" << endl;
40                             return false;
41                     }
42                     strcpy(newnode->content, input);
43                     newnode->count = 1;
44                     nonodes = 1;
45                     newnode->next = NULL;
46                     head = newnode;
47
48             }
49             else            // if not here then the linked list exists
50             {
51                     NPtr check = head;
52                     while (check != NULL)
53                     {
54                             if (strcmp(check->content, input) == 0)
55                             {
56                                     check->count++;
```

Oct 25 11:22 2014 list.cpp Page 2

```
 57                                      break;
 58                              }
 59                              if (check->next == NULL)
 60                              {
 61                                      NPtr newnode = new node;
 62                                      if (newnode == NULL)
 63                                      {
 64                                              cout << "Allocation error occured" << endl;
 65                                              return false;
 66                                      }
 67                                      strcpy(newnode->content, input);
 68                                      newnode->count++;
 69                                      nonodes += 1;
 70                                      newnode->next = NULL;
 71                                      check->next = newnode;
 72                                      break;
 73                              }
 74                              check = check->next;
 75
 76                      }
 77              }
 78              return true;
 79
 80      }
 81
 82      bool LIST::print(ostream&)
 83      {
 84              if (head == NULL)
 85              {
 86                      return false;
 87              }
 88              NPtr temp = head;                               // do not change the head pointer
 89              while (temp != NULL)
 90              {
 91                      cout << temp->content << '\t';
 92                      cout << temp->count;
 93                      cout << endl;
 94                      temp = temp->next;
 95              }
 96              return true;
 97
 98      }
 99
100      NPtr LIST::returnhead()
101      {
102              return head;
103      }
```

Oct 25 11:22 2014 main.cpp Page 1

```
 1     /*******************************
 2     *Filename:main.cpp              *
 3     *Login:by932                   *
 4     *AssignmentNo:ass4             *
 5     *DateLastModified:19/10/2014   *
 6     *******************************/
 7     #include <iostream>
 8     #include <fstream>
 9     #include <cstring>
10     #include "hash.h"
11     using namespace std;
12
13     int main()
14     {
15             int file_len = 0;
16             char source_file[20]= {'\0'};
17             char data[21] = {'\0'};
18             HT start;
19
20             cout << "Enter file name? ";
21             cin >> source_file;
22
23             file_len = strlen(source_file);
24             ifstream read;
25             read.open(source_file);
26
27             if(!read)
28             {
29                     cout << "Cann't find file"<< endl;
30                     return 0;
31             }
32             else if (strcmp(source_file + file_len - 4, ".fna") == 0)
33             {
34                     char del[1025] = {'\0'};
35
36                     cout << "This file is a fna file" << endl;
37                     read.getline (del,1024,'\n');
38                     while(!read.eof())
39                     {
40                             for (int i = 0; i < 21; i++)
41                                     data[i] = '\0';
42
43                             for (int i = 0; i < 8; i++)
44                             {
45                                     data[i] = read.get();
46                                     while (data[i] == ' ' || data[i] == '\n')
47                                     {
48                                             data[i] = read.get();
49                                     }
50
51                             }
52                             if (!read.good())
53                             {
54                                     read.ignore();
55                                     read.clear();
56                                     break;
```

Oct 25 11:22 2014 main.cpp Page 2

```
 57                              }
 58                              start.calculate(data);
 59                      }
 60              }
 61              else if (strcmp(source_file + file_len - 4, ".txt") == 0)
 62              {
 63                      cout << "This file is a txt file" << endl;
 64
 65                      while(!read.eof())
 66                      {
 67                              for (int i = 0; i < 21; i++)
 68                                      data[i] = '\0';
 69
 70                              for (int i = 0; i < 21; i++)
 71                              {
 72                                      data[i] = read.get();
 73
 74                                      if (data[i] == ' ' || data[i] == '\n')
 75                                      {
 76                                              data[i] = '\0';
 77                                              break;
 78                                      }
 79
 80                              }
 81                              if (!read.good())
 82                              {
 83                                      read.ignore();
 84                                      read.clear();
 85                                      break;
 86                              }
 87                              start.calculate(data);
 88                      }
 89              }
 90              else
 91              {
 92                      cout << "This file is not a txt file OR fna file" << endl;
 93                      return 0;
 94              }
 95              start.findtop();
 96
 97              read.close();
 98              return 0;
 99      }
100
```

Oct 25 11:22 2014 constants.h Page 1

```
 1     /*******************************
 2     *Filename:constants.h           *
 3     *Login:by932                    *
 4     *AssignmentNo:ass4              *
 5     *DateLastModified:19/10/2014    *
 6     *******************************/
 7
 8
 9     const int SIZE = 100;   //how bigger hash table
10     struct sortnode
11     {
12             char content[21];
13             int count;
14     };
15
```

Oct 25 11:22 2014 hash.h Page 1

```
 1     /*******************************
 2     *Filename:hash.h                *
 3     *Login:by932                    *
 4     *AssignmentNo:ass4              *
 5     *DateLastModified:19/10/2014    *
 6     *******************************/
 7     #include <iostream>
 8     #include <cstring>
 9     #include "list.h"
10     using namespace std;
11
12     struct htnode
13     {
14        LIST data;
15     };
16
17     typedef htnode* HDNPtr;
18
19     class HT
20     {
21             public:
22                     HT();
23                     ~HT();
24                     void calculate(char input[]);
25                     void findtop();
26                     bool topten(LIST&);
27                     void sort(sortnode[]);
28             private:
29                     HDNPtr hashtable[SIZE];
30                     sortnode sortarr[10];
31     };
32
```

Oct 25 11:22 2014 list.h Page 1

```
 1     /*******************************
 2     *Filename:list.h                *
 3     *Login:by932                    *
 4     *AssignmentNo:ass4              *
 5     *DateLastModified:19/10/2014    *
 6     *******************************/
 7     #include <iostream>
 8     #include "constants.h"
 9     #include <cstring>
10     using namespace std;
11     struct node
12     {
13             char content[21];
14             int count;
15             node *next;
16     };
17     typedef node* NPtr;
18     class LIST
19     {
20             public:
21                     LIST();
22                     LIST(const LIST&);
23                     ~LIST();
24                     bool load(char input[]);
25                     bool print(ostream&);
26                     NPtr returnhead();
27             private:
28                     int nonodes;    // no of nodes in LIST
29                     NPtr head;    // pointer to list of chars
30                     NPtr next;
31     };
32
33
34
```

```
Thanks for submitting constants.h
Thanks for submitting list.h
Thanks for submitting list.cpp
Thanks for submitting hash.h
Thanks for submitting hash.cpp
Thanks for submitting main.cpp

Compiling main.cpp hash.cpp list.cpp

g++ main.cpp hash.cpp list.cpp
```