|  |
| --- |
| List.h  #pragma once  #include<iostream>  using namespace std;  class list  {  int \*a;  int max;  int size, ci;  public:  list(void)  {  a = nullptr;  max = 0;  size = 0;  ci = 0;  }  ~list(void)  {  }  void createlist();  void deallocate();  void insertion();  void deletion();  void update();  void start();  void next();  void back();  void tail();  int find();  void copy();  void get();  void length();  void displaylist();  void clear();  void countzero();  void exit();  }; |

|  |
| --- |
| List.cpp  #include "list.h"  #include<iostream>  using namespace std;  int main()  {  list l;  int c, f;  do  {  cout << "1: create list" << endl  << "2: insertion" << endl  << "3: deletion" << endl  << "4: update" << endl  << "5: start" << endl  << "6: next" << endl  << "7: back" << endl  << "8: tail" << endl  << "9: find" << endl  << "10: copy" << endl  << "11: get" << endl  << "12: length" << endl  << "13: display list" << endl  << "14: count no of zeros" << endl  << "15: deallocate list and clear" << endl  << "16: exit" << endl  << "enter your choice:";  cin >> c;  switch (c)  {  case 1:  {  l.createlist();  break;  }  case 2:  {  l.insertion();  break;  }  case 3:  {  l.deletion();  break;  }  case 4:  {  l.update();  break;  }  case 5:  {  l.start();  break;  }  case 6:  {  l.next();  break;  }  case 7:  {  l.back();  break;  }  case 8:  {  l.tail();  break;  }  case 9:  {  f = l.find();  if (f == 0)  {  cout << "error: no not found" << endl;  }  else  {  cout << "no successfully found" << endl;  }  break;  }  case 10:  {  l.copy();  break;  }  case 11:  {  l.get();  break;  }  case 12:  {  l.length();  break;  }  case 13:  {  l.displaylist();  break;  }  case 14:  {  l.countzero();  }  case 15:  {  l.deallocate();  break;  }  case 16:  {  l.exit();  break;  }  default:cout << "enter correct choice" << endl;  }  } while (c != 16);  system("pause");  return 0;  }  void list::createlist()  {  cout << "enter the size of array";  cin >> max;  a = new int[max + 1];  }  void list::deallocate()  {  clear();  delete[]a;  }  void list::insertion()  {  int x, f;  char c;  if (max != 0)  {  if (size == 0)  {  cout << "how many values you want to enter: ";  cin >> size;  if (size <= max)  {  for (int i = 1; i <= size; i++)  {  cin >> a[i];  ci = i;  }  }  else  cout << "values amount must be smaller then the size of an array" << endl;  }  else  {  do  {  cout << "a: by location" << endl  << "b: by value" << endl  << "n:exit" << endl;  cin >> c;  switch (c)  {  case 'a':  {  cout << "1: After the current location"<<endl  << "2: Before the current location"<<endl  << "3: At the current location"<<endl  << "4: exit" << endl;  cin >> x;  switch (x)  {  case 1:  {  for (int i = size; i > ci; i--)  {  a[i + 1] = a[i];  }  cout << "enter a new value :";  cin >> a[ci + 1];  size++;  break;  }  case 2:  {  for (int i = size; i >= (ci - 1); i--)  {  a[i + 1] = a[i];  }  cout << "enter a new value: ";  cin >> a[ci - 1];  size++;  break;  }  case 3:  {  for (int i = size; i > ci; i--)  {  a[i + 1] = a[i];  }  cout << "enter a new value :";  cin >> a[ci];  size++;  break;  }  case 4:  {  break;  }  default:cout << "enter correct choice" << endl;  }  break;  }  case 'b':  {  cout << "1: After the value"<<endl  << "2: Before the value"<<endl  << "3: exit" << endl;  switch (x)  {  case 1:  {  f = find();  if (f != 0)  {  for (int i = size; i > ci; i--)  {  a[i + 1] = a[i];  }  cout << "enter a new value :";  cin >> a[ci + 1];  size++;  }  else  {  cout << "error" << endl;  }  break;  }  case 2:  {  f = find();  if (f != 0)  {  for (int i = size; i >= (ci - 1); i--)  {  a[i + 1] = a[i];  }  cout << "enter a new value: ";  cin >> a[ci - 1];  size++;  }  else  {  cout << "error" << endl;  }  break;  }  case 3:  {  break;  }  default:cout << "enter correct choice" << endl;  }  break;  }  default:cout << "enter correct choice" << endl;  }  } while (c != 'n');  }  }  cout << "error" << endl;  }  void list::deletion()  {  int c, f, x, y = 0, prime = 0;  if (size != 0)  {  cout << "1: By Current location" << endl  << "2: by value" << endl  << "3: odd/even no" << endl  << "4: prime" << endl  << "5:exit" << endl  << "enter your choice";  cin >> c;  switch (c)  {  case 1:  {  for (int i = ci; i <= size; i++)  {  a[i] = a[i + 1];  }  size--;  }  case 2:  {  f = find();  if (f != 0)  {  for (int i = ci; i <= size; i++)  {  a[i] = a[i + 1];  }  size--;  }  else  {  cout << "error: number not found" << endl;  }  }  case 3:  {  cout << "1: even" << endl  << "2: odd" << endl;  cin >> x;  for (int i = 1; i <= size; i++)  {  if (x == 1)  {  if (a[i] % 2 == 0)  {  for (int i = ci; i <= size; i++)  {  a[i] = a[i + 1];  }  size--;  }  }  else if (x == 2)  {  if (a[i] % 2 != 0)  {  for (int i = ci; i <= size; i++)  {  a[i] = a[i + 1];  }  size--;  }  }  else  cout << "enter correct choice" << endl;  }  case 4:  {  for (int i = 1; i < size; i++)  {  for (int j = 2; j < a[i]; j++)  {  if (a[i] % j == 0)  {  y = 1;  }  }  if (y == 0)  {  prime++;  for (int i = ci; i <= size; i++)  {  a[i] = a[i + 1];  }  size--;  }  }  if (prime == 0)  cout << "there is no prime number in array" << endl;  }  }  default:cout << "enter correct choice" << endl;  }  }  else  {  cout << "error" << endl;  }  }  void list::update()  {  int flag = find();  if (flag != 0)  {  cout << "enter value to update";  cin >> a[ci];  }  else  {  cout << "error: number not found" << endl;  }  }  void list::start()  {  cout << "start" << endl;  ci = 1;  cout << "index of start: " << ci << endl;  }  void list::next()  {  cout << "next" << endl;  if (ci != size)  {  ci++;  cout << "next index: " << ci << endl;  }  else  cout << "error" << endl;  }  void list::back()  {  cout << "back" << endl;  if (ci != 1)  {  ci--;  cout << "back index: " << ci << endl;  }  else  cout << "error" << endl;  }  void list::tail()  {  cout << "tail" << endl;  ci = size;  cout << "tail index: " << ci << endl;  }  int list::find()  {  cout << "find" << endl;  int x, c = 0;  cout << "enter the value to find: ";  cin >> x;  for (int i = 1; i <= size; i++)  {  if (a[i] == x)  {  ci = i;  c = 1;  break;  }  }  return c;  }  void list::copy()  {  cout << "copy" << endl;  }  void list::get()  {  cout << "get" << endl;  cout << "current index is " << ci << " and its value is " << a[ci] << endl;  }  void list::length()  {  cout << "length" << endl;  cout << "total elements are: " << size << endl;  }  void list::displaylist()  {  cout << "display list" << endl;  for (int i = 1; i <= size; i++)  cout << i << " element of list: " << a[i] << endl;  }  void list::clear()  {  cout << "clear" << endl;  size = 0;  ci = 0;  }  void list::countzero()  {  int x = 0;  for (int i = 1; i <= size; i++)  {  if (a[i] == 0)  x++;  }  if (x != 0)  cout << "total zeros in array: " << x << endl;  else  cout << "there is no zero in an array" << endl;  }  void list::exit()  {  cout << "exit" << endl;  cout << "allah hafiz" << endl;  deallocate();  } |