Đề 1:

1. Viết chương trình tính tổng các chữ số n ( n > 10^16)
2. Viết hàm tính tổng các số nguyên tố trong mảng 1 chiều
3. Đếm các từ xuất hiện nhiều hơn k lần trong 1 chuỗi
4. Cấu trúc thời gian: 3 thông số giờ, phút, giây, thuộc kiểu int, thực hiện tính thời điểm sau n (giây) từ 1 mốc thời gian cho trước.
5. Viết hàm xóa các nút đứng trước nút có giá trị k.

Đề 2:

1. Tìm n số fibonaci đầu tiên (n > 2) biết a[1] = 1, a[2] = 1, …, a[n] = a[n-1] + a[n-2]
2. Viết hàm tính tổng các phần tử trên đường chéo phụ ma trận
3. Chuẩn hóa xâu theo đúng cú pháp (Viết hoa đầu dòng, đầu câu, mỗi từ cách nhau 1 khoảng trắng)
4. Cấu trúc sinh viên gồm: Họ tên (string), điểm (float), học lực (string), hãy nhập n sinh viên từ bàn phím, tiến hành phân loại học lực theo thang điểm
   1. Điểm >= 8.5 : Gioi
   2. Điểm >= 6 : Kha
   3. Điểm >= 4 : Trung Binh
   4. Điểm < 4 : Yeu

Thực hiện xuất ra danh sách sinh viên học lực Trung Binh

1. Viết hàm sắp xếp 1 danh sách theo thứ tự giảm dần

|  |
| --- |
| /\* |
|  | // B1 |
|  | #include <iostream> |
|  | #include <string.h> |
|  |  |
|  | using namespace std; |
|  |  |
|  | int main(){ |
|  | string a; |
|  | cin >> a; |
|  | int sum = 0; |
|  | for(int i=0; i<a.size(); i++) |
|  | sum += a[i] - '0'; |
|  | cout << sum; |
|  | } |
|  |  |
|  | // B2: |
|  | #include <iostream> |
|  | #include <math.h> |
|  |  |
|  | using namespace std; |
|  |  |
|  | int KTNT(int n){ |
|  | if(n < 2) return 0; |
|  | for(int i=2; i<= sqrt(n); i++) |
|  | if(n % i == 0) return 0; |
|  | return 1; |
|  | } |
|  | int SumNT(int \*a, int n){ |
|  | int sum = 0; |
|  | for(int i=0; i<n; i++) |
|  | if(KTNT(a[i]) == 1) |
|  | sum += a[i]; |
|  | return sum; |
|  | } |
|  | int main(){ |
|  | int n; |
|  | cout << "Nhap so luong: "; cin >> n; |
|  | int \*a = new int[n]; |
|  | cout << "Nhap mang:\n"; |
|  | for(int i=0; i<n; i++) |
|  | cin >> a[i]; |
|  | cout << "Tong cac so nguyen to trong mang: " << SumNT(a, n); |
|  | } |
|  |  |
|  | // B3 |
|  | #include <iostream> |
|  | #include <string.h> |
|  | #include <vector> |
|  |  |
|  | using namespace std; |
|  | vector <string> TachTu(string s){ |
|  | vector <string> Kq; |
|  | int i; |
|  | while(s.size() != 0){ |
|  | for(i = 0; i<=s.size(); i++) |
|  | if(s[i] == 32 || i == s.size()){ |
|  | string x = s.substr(0, i); |
|  | Kq.push\_back(x); |
|  | s.erase(0, i + 1); |
|  | break; |
|  | } |
|  | } |
|  | return Kq; |
|  | } |
|  | int Dem(string s, int k){ |
|  | vector <string> tu = TachTu(s); |
|  | int kq = 0, count; |
|  | string Xoa = " "; |
|  | for(int i=0; i<tu.size(); i++) |
|  | if(Xoa.compare(tu[i]) != 0){ |
|  | count = 1; |
|  | for(int j = i+1; j<tu.size(); j++) |
|  | if(tu[i].compare(tu[j]) == 0){ |
|  | count++; |
|  | tu[j] = Xoa; |
|  | } |
|  | if(count >= k) |
|  | kq++; |
|  | } |
|  | return kq; |
|  | } |
|  | int main(){ |
|  | string s; |
|  | int k; |
|  | cout << "Nhap chuoi: "; getline(cin, s); |
|  | cout << "Nhap k: "; cin >> k; |
|  | vector <string> Kq = TachTu(s); |
|  | cout << Dem(s, k); |
|  |  |
|  | } |
|  | // B4 |
|  |  |
|  | #include <iostream> |
|  |  |
|  | using namespace std; |
|  |  |
|  | typedef struct Time{ |
|  | int h; |
|  | int p; |
|  | int s; |
|  | }T; |
|  | void Nhap(Time &a){ |
|  | char c; |
|  | cout << "Nhap moc thoi gian(h:p:s): "; cin >> a.h >> c >> a.p >> c >> a.s; |
|  | } |
|  | void ChuanHoa(Time &a){ |
|  | if(a.s >= 60){ |
|  | a.p += a.s/60; |
|  | a.s %= 60; |
|  | } |
|  | if(a.p >= 60){ |
|  | a.h += a.p/60; |
|  | a.p %= 60; |
|  | } |
|  | if(a.h > 23) |
|  | a.h %= 23; |
|  | } |
|  | void Xuat(Time a){ |
|  | if(a.h<10) cout << "0"; |
|  | cout << a.h << ":"; |
|  | if(a.p < 10) cout << "0"; |
|  | cout << a.p << ":"; |
|  | if(a.s < 10) cout << "0"; |
|  | cout << a.s; |
|  | } |
|  | Time thoiDiemSau(Time a, int n){ |
|  | Time kq = a; |
|  | kq.s += n; |
|  | ChuanHoa(kq); |
|  | return kq; |
|  | } |
|  | int main(){ |
|  | Time trc; |
|  | Nhap(trc); |
|  | int n; |
|  | cout << "Nhap n(s): "; cin >> n; |
|  | ChuanHoa(trc); |
|  | cout << "Thoi diem ban dau: "; Xuat(trc); |
|  | cout << "\nSau " << n << " (s) la: "; Xuat(thoiDiemSau(trc, n)); |
|  |  |
|  | } |
|  |  |
|  | // B5: |
|  |  |
|  |  |
|  | #include <iostream> |
|  |  |
|  | using namespace std; |
|  | struct Node{ |
|  | int data; |
|  | Node\* next; |
|  | }; |
|  | Node \*KhoiTao(int x){ |
|  | Node \*p = new Node; |
|  | p->data = x; |
|  | p->next = NULL; |
|  | return p; |
|  | } |
|  | Node \*addDau(Node \*head, int x){ |
|  | Node \*p = KhoiTao(x); |
|  | p->next = head; |
|  | return p; |
|  | } |
|  | void Xuat(Node \*head){ |
|  | while(head != NULL){ |
|  | cout << head->data << " "; |
|  | head = head->next; |
|  | } |
|  | } |
|  | Node\* Delete(Node \*head, int k){ |
|  | while(head->next->data == k && head->data != k){ |
|  | head = head->next; |
|  | if(head->next == NULL) |
|  | break; |
|  | } |
|  | Node \*i = head; |
|  | while(i->next != NULL){ |
|  | if(i->next->data != k && i->next->next->data == k){ |
|  | i->next = i->next->next; |
|  | } |
|  | i = i->next; |
|  | } |
|  | return head; |
|  | } |
|  | int main(){ |
|  | Node \*head = NULL; |
|  | int n, k; |
|  | cout << "Nhap n: "; cin >> n; |
|  | cout << "Nhap gtri:\n"; |
|  | for(int i=0; i<n; i++){ |
|  | int x; cin >> x; |
|  | head = addDau(head, x); |
|  | } |
|  | cout << "Ban dau: "; Xuat(head); |
|  | cout << "\nNhap k: "; cin >> k; |
|  | head = Delete(head, k); |
|  | cout << "Sau: "; Xuat(head); |
|  | } |
|  |  |