

3057

# BÀI BÁO CÁO ĐỒ ÁN TIN HỌC 2022

**Giáo viên phụ trách: Trần Thị Mỹ Huỳnh**

**Tên sinh viên: Nguyễn Bùi Nhựt Ý**

**MSSV: DH52006931**

**Lớp: D20\_Th11**




Logo  
Name

---

## LỜI CẢM ƠN

Em xin chân thành cảm ơn Trường Đại Học Công Nghệ Sài Gòn đã tạo điều kiện cho chúng em học tập và thực hiện bản báo cáo đề án môn học này. Em xin bày tỏ lòng biết ơn sâu sắc tới đã tận tình hướng dẫn chỉ bảo em trong quá trình thực hiện đề tài. Em xin chân thành cảm ơn cô trong khoa Công nghệ Thông tin đã tận tình giảng dạy, và trang bị cho em những kiến thức có ích, cần thiết trong suốt quá trình học tập tại khoa. Trong suốt quá trình thực hiện đề tài, mặc dù đã cố gắng hết sức để hoàn thành nhưng chắc chắn bài báo cáo không tránh khỏi những thiếu sót, kính mong cô tận tình chỉ bảo. Một lần nữa, em xin gửi lời cảm ơn chân thành đến tất cả mọi người, những ai đã quan tâm, giúp đỡ và ủng hộ em trong suốt thời gian qua. Cảm ơn mọi người đã chú ý lắng nghe.



---

# Đề 1: Quản lý siêu thị bằng Simple Linked List

**Câu 1: Nhập danh sách siêu thị với đầy đủ thông tin**

**Câu 2: Xuất danh sách siêu thị với đầy đủ các thông tin**

**Câu 3: Nhập vào 1 mã siêu thị, kiểm tra mã siêu thị đó có tồn tại? Nếu có thì xuất ra màn hình thông tin**

**Câu 4: In ra màn hình thông tin Siêu thị có doanh số lớn nhất và nhỏ nhất**

**Câu 5: In ra màn hình thông tin Siêu thị có số lượng nhân viên lớn nhất và nhỏ nhất**

**Câu 6: Xuất ra File Text thông kê số lượng nhân viên dưới 100, trên 200 và 100-200**

**Câu 7: Tính doanh số của siêu thị (DS) từ năm thành lập đến bây giờ (Year)**

---

**Câu 8: Nhập lương cơ bản của nhân viên trong 1 năm (LCB). Tính lợi nhuận của siêu thị trong 1 năm (LN)**

**$DS = \text{Doanh số cơ bản} * (\text{Year} - \text{Năm thành lập})$**

**Câu 9: Nhập vào 1 con số lợi nhuận (SLN) và liệt kê các siêu thị có lợi nhuận > SLN**

**$LN = \text{Doanh số cơ bản} - (\text{LCB} * \text{Số lượng nhân viên})$**

**Câu 10: Sắp xếp danh sách tăng dần theo năm thành lập**

## File Header.h

```
1  #pragma once
2
3  #define _CRT_SECURE_NO_WARNINGS
4
5  #include <iostream>
6  #include <iomanip>
7  #include <string.h>
8  #include <fstream>
9  #include <Windows.h>
10
11  using namespace std;
12
13  struct st
14  {
15      char mst[50]; //na sieu thi
16      char tst[50]; // ten sieu thi
17      int ntl; // nam thanh lap
18      int dscb; // danh so co ban ( trung binh trong 1 nam
19      int slnv; // so luong nhan vien
20  };
21
22  struct QLST
23  {
24      st data;
25      QLST* link;
26  };
27
28  typedef struct QLST* qlst;
```

```

30 void Nhap1ST(st& st);
31 void KhoiTao(qlst& s);
32 int IsEmpty(qlst s);
33 void Giaiphong(qlst& s);
34 qlst CreatNode(st x);
35 qlst Them(qlst& s, st x);
36 bool KT_TrungMa(qlst& s, char* mst);
37 void XuatDSST(qlst s);
38 void KhungXuat();
39 void Xuat1ST(st s);
40 void NhapDSST(qlst& list);
41 void KT_MST(qlst s, char mstkt[50]);
42 void MinDSCB(qlst s);
43 void MaxDSCB(qlst s);
44 void MinSLNV(qlst s);
45 void MaxSLNV(qlst s);
46 int Xuat_File_TK_SLNV(qlst s);
47 void DoanhSo(qlst s, char mstds[50]);
48 int LoiNhuan(qlst s, int lcb, char mstds[50]);
49 void ListsWithGreaterProfits(qlst s);
50 void Swap(qlst a, qlst b);
51 void DSSTangDan(qlst& s);
52 void Menu();
53 int CountNode(qlst s);
54 void DSMenu(qlst s);

```

## File Main.cpp

```

1  #include "Header.h"
2
3  int main() {
4      SetConsoleTitle(L"Do An Tin Hoc: SIMPLE LINKED LIST QUAN LY DANH SACH SIEU THI");
5      cout << "\nDo an Tin Hoc: SIMPLE LINKED LIST QUAN LY DANH SACH SIEU THI\n";
6      system("color 2");
7      qlst s;
8      KhoiTao(s);
9      DSMenu(s);
10     return 0;

```

## File Source.cpp Gồm:

## Hàm Khung Menu:

[illegible]

## Hàm khởi tạo list (s):

```

2
3 void KhoiTao(qlst& s) {
4     s = NULL;
5 }

```

### Hàm kiểu int kiểm tra xem List(s) có rỗng ko:

```
7 int IsEmpty(qlst s) {
8     return s == NULL ? 1 : 0;
9 }
```

Dòng 8: Dùng toán tử 3 khi so sánh List(s) có bằng NULL không? Có(1), Không (0).

### Hàm giải phóng Node trong List (s):

```
11 void GiaiPhong(qlst& s) {
12     qlst node = s;
13     while (node != NULL) {
14         s = s->link;
15         delete node;
16         node = s;
17     }
18 }
```

Dòng 12: Cho con trỏ node gán bằng s.

### Dòng 13 - 16:

- Dừng vòng lặp while điều kiện chạy khi node khác NULL.
- S lúc này sẽ cập nhật về đằng sau của nó.
- Dừng lệnh xóa node.
- Và cập nhật node sang s.

## Hàm khởi tạo 1 Node trong List:

```
20 qlst CreatNode(st x)
21 {
22     qlst node = new QLST;
23     node->data = x;
24     node->link = NULL;
25     return node;
26 }
```

Dòng 22: Khởi tạo node.

Dòng 23: Data gán bằng x ( x là dữ liệu 1 siêu thị ).

## Hàm thêm Node(x) vào List (s):

```
28 qlst Them(qlst& s, st x) {
29     qlst p = CreatNode(x);
30     p->link = s;
31     s = p;
32     return p;
33 }
```

Dòng 29: Khởi tạo p.

Dòng 30: P sẽ trở vào s.

Dòng 31: Cập nhật s sang p.

## Hàm kiểu bool kiểm tra xem có trùng mã không:

```
35 bool KT_TrungMa(qlst& s, char* mst) {
36     qlst node = s;
37     while (node != NULL) {
38         if (strcmp(node->data.mst, mst) == 0) return true;
39         node = node->link;
40     } return false;
41 }
```

Dòng 36: Cho node gán bằng s.

Dòng 37 - 40:

- Dùng vòng lặp while điều kiện chạy khi Node khác NULL.
- Kiểm tra chuỗi giữa data mã siêu thị trong node đang hiện có và mã siêu thị cần tìm bằng nhau thông qua hàm strcmp thì trả về True còn không thì False.



## Câu 1: Nhập danh sách siêu thị với đầy đủ thông tin: Hàm nhập 1 siêu thị:

```
44 void Nhap1ST(st& st)
45 {
46     time_t t = time(0);
47     cin.ignore();
48     cout << "\nNhap ten sieu thi: ";
49     cin.getline(st.tst, 30);
50     do
51     {
52         cout << "\nNhap nam thanh lap: ";
53         cin >> st.ntl;
54         if (st.ntl < 0 || st.ntl > (localtime(&t)->tm_year + 1900))
55         {
56             cout << "\nNhap nam thanh lap qua nam hien tai or la am";
57             cout << "\nVui long nhap lai";
58         }
59     } while (st.ntl < 0 || st.ntl > (localtime(&t)->tm_year + 1900));
60     do
61     {
62         cout << "\nNhap doanh so co ban: ";
63         cin >> st.dscb;
64         if (st.dscb < 0)
65         {
66             cout << "\nNhap doanh so co ban am";
67             cout << "\nVui long nhap lai";
68         }
69     } while (st.dscb < 0);
70     do
71     {
72         cout << "\nNhap so luong nhan vien: ";
73         cin >> st.slnv;
74         if (st.slnv < 0)
75         {
76             cout << "\nNhap so luong nhan vien am";
77             cout << "\nVui long nhap lai";
78         }
79     } while (st.slnv < 0);
80     cin.ignore();
81     system("pause");
82 };
```

Dòng 46: Lấy type time trong thư viện Time gán bằng dữ liệu thời gian hiện tại.

để khi tới ràng buộc dòng 59 thì check xem năm thành lập có lớn hơn năm hiện tại không. Để khỏi phải bảo trì theo từng năm tự động hóa, ta dùng `localtime(&t)->tm_year + 1900`.

Dòng 47 & 80: Loại bỏ các ký tự, tối đa số được chỉ định hoặc cho đến khi đạt đến dấu phân cách.

Dòng 48 - 79: Nhập dữ liệu siêu thị bao gồm (Tên Siêu Thị, Năm Thành Lập, Doanh Số Cơ Bản, Số Lượng Nhân Viên ). Khi lúc nhập gần từng ràng buộc.

Dòng 81: Dừng hệ thống khi nhập xong 1 siêu thị.

## Hàm nhập nguyên danh sách siêu thị

```
84 void NhapDSST(qlst& list) {
85
86     st st;
87     while (true)
88     {
89         cout << "\nNhap thong tin 1 sieu thi: ";
90         cout << "\nNhap ma sieu thi (Nhap -99 de dung): ";
91         cin >> st.mst;
92         while (KT_TrungMa(list, st.mst))
93         {
94             cout << "\nNhap lai ma sieu thi (Trung ma): ";
95             cin >> st.mst;
96         }
97         if (strcmp(st.mst, "-99") == 0) break;
98         Nhap1ST(st);
99         Them(list, st);
100     }
101 }
```

Dòng 86 - 99:

- Gọi struct st.
- Dùng vòng lặp white điều kiện chạy true ( vô tận, dừng lại khi ta nhập Mã Siêu thị = -99 ở dòng 97 ).
- Nhập thêm Mã Siêu Thị để check xem mã có bị trùng không khi gọi lại hàm KT\_TrungMa (list, st.mst) trong điều kiện chạy của while .
- Sau đó Gọi hàm Nhap1ST (st).
- Và thêm siêu thị ấy vào List.

**Giả sử: ta nhập lần lượt dữ liệu là 5, 3 9 tương ứng cho từng thông tin của siêu thị.**

**Kết quả:**

```
Do An Tin Hoc: SIMPLE LINKED LIST QUAN LY DANH SACH SIÊU THỊ

Nhap thong tin 1 siêu thị:
Nhap ma siêu thị (Nhap -99 de dung): 5

Nhap ten siêu thị: 5

Nhap nam thanh lap: 5

Nhap doanh so co ban: 5

Nhap so luong nhan vien: 5
Press any key to continue . . .

Nhap thong tin 1 siêu thị:
Nhap ma siêu thị (Nhap -99 de dung): 3

Nhap ten siêu thị: 3

Nhap nam thanh lap: 3

Nhap doanh so co ban: 3

Nhap so luong nhan vien: 3
Press any key to continue . . .

Nhap thong tin 1 siêu thị:
Nhap ma siêu thị (Nhap -99 de dung): 9

Nhap ten siêu thị: 9
```

**Kết quả khi viết trùng mã:**

```
Nhap thong tin 1 siêu thị:
Nhap ma siêu thị (Nhap -99 de dung): 3

Nhap lai ma siêu thị (Trung ma): 3
```

**Kết quả khi ta nhập mã = -99:**  
**Trở về Menu.**

```
Nhap lai ma sieu thi (Trung ma): -99
=====MENU=====
0. Thoat!!!
1. Nhap danh sach sieu thi voi day du cac thong tin
2. Xuat danh sach sieu thi voi day du cac thong tin
3. Nhap vao 1 ma sieu thi vakiem tra ma sieu thi do co ton tai? Neu co thi xuat ra man hinh thong tin
4. In ra man hinh thong tin sieu thi co doanh so lon nhat
5. In ra man hinh thong tin sieu thi co doanh so nho nhat
6. In ra man hinh thong tin sieu thi co so luong nhan vien lon nhat
7. In ra man hinh thong tin sieu thi co so luong nhan vien nho nhat
8. Xuat ra File Text: duoi 100 nhan vien, 100-200 nhan vien, tren 200 nhan vien
9. Tinh doanh so cua sieu thi (DS) tu nam thanh lap den bay gio (Year)
10. Nhap luong co ban cua nhan vien trong 1 nam (LCB). Tinh loi nhuan cua sieu thi trong 1 nam (LN)
    DS = Doanh so co ban * (Year - Nam thanh lap)
11. Nhap vao 1 con so loi nhuan (SLN) va liet ke cac sieu thi co loi nhuan > SLN
    LN = Doanh so co ban - (LCB * So luong nhan vien)
12. Sap xep danh sach tang dan theo nam thanh lap
=====MENU=====
```

**Câu 2: Xuất danh sách siêu thị với đầy đủ các thông tin:**  
**Hàm xuất danh sách siêu thị:**

```
103 //Cau 2
104 void XuatDSST(qList s)
105 {
106     if (!IsEmpty(s))
107     {
108         cout << "\nDanh sach sieu thi\n";
109         KhungXuat();
110         qList node = s;
111         while (node != NULL)
112         {
113             Xuat1ST(node->data);
114             node = node->link;
115         }
116     } else cout << "\nDanh sach sieu thi rong\n";
117 }
```

Dòng 106 – 116:

- Nếu List khác NULL thì xuất khung ra trước.
- Cho node gán bằng List (s).
- Dùng duyệt để xuất từng siêu thị chứa node data đi qua.
- Nếu List bằng NULL thì xuất văn bản ra màn hình là Danh sách siêu thị rỗng.

## Hàm khung xuất:

```
119 void KhungXuat()  
120 {  
121     cout << endl << "Ma sieu thi"  
122         << setw(20) << left << "Ten sieu thi"  
123         << setw(20) << left << "Nam thanh lap"  
124         << setw(20) << left << "Doanh thu co ban"  
125         << setw(20) << left << "So luong nhan vien"  
126         << endl;  
127 }
```

Dùng setw() cho số ô dữ liệu tính từ trái qua.

## Hàm xuất 1 siêu thị:

```
129 void Xuat1ST(st s)  
130 {  
131     cout << endl  
132         << setw(20) << left << s.mst  
133         << setw(20) << left << s.tst  
134         << setw(15) << left << s.ntl  
135         << setw(20) << left << s.dscb  
136         << setw(20) << left << s.slrv  
137         << endl;  
138 }
```

## Kết quả:

```
Nhap lua chon: 2  
Danh sach sieu thi  
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban    So luong nhan vien  
9          9                9                  9                  9  
3          3                3                  3                  3  
5          5                5                  5                  5
```

## Kết quả khi ta không nhập dữ liệu xuất làm danh sách rỗng:

```
Do An Tin Hoc: SIMPLE LINKED LIST QUAN LY DANH SACH SIEU THI

Do an Tin Hoc: SIMPLE LINKED LIST QUAN LY DANH SACH SIEU THI

=====MENU=====
0. Thoat!!!
1. Nhap danh sach sieu thi voi day du cac thong tin
2. Xuat danh sach sieu thi voi day du cac thong tin
3. Nhap vao 1 ma sieu thi vakiem tra ma sieu thi do co ton tai? Neu co thi xuat ra man hinh thong tin
4. In ra man hinh thong tin sieu thi co doanh so lon nhat
5. In ra man hinh thong tin sieu thi co doanh so nho nhat
6. In ra man hinh thong tin sieu thi co so luong nhan vien lon nhat
7. In ra man hinh thong tin sieu thi co so luong nhan vien nho nhat
8. Xuat ra File Text: duoi 100 nhan vien, 100-200 nhan vien, tren 200 nhan vien
9. Tinh doanh so cua sieu thi (DS) tu nam thanh lap den bay gio (Year)
10. Nhap luong co ban cua nhan vien trong 1 nam (LCB). Tinh loi nhuan cua sieu thi trong 1 nam (LN)
    DS = Doanh so co ban * (Year - Nam thanh lap)
11. Nhap vao 1 con so loi nhuan (SLN) va liet ke cac sieu thi co loi nhuan > SLN
    LN = Doanh so co ban - (LCB * So luong nhan vien)
12. Sap xep danh sach tang dan theo nam thanh lap
=====MENU=====

Nhap lua chon: 2
Danh sach sieu thi rong
```

**Câu 3: Nhập vào 1 mã siêu thị, kiểm tra mã siêu thị đó có tồn tại? Nếu có thì xuất ra màn hình thông tin:**

**Hàm kiểm tra xem mã cần tìm có trong List không:**

```
140 //Cau 3
141 void KT_MST(qlst s, char mstkt[50])
142 {
143     if (!IsEmpty(s))
144     {
145         bool flag = false;
146         do {
147             if (strcmp(s->data.mst, mstkt) == 0)
148             {
149                 KhungXuat();
150                 Xuat1ST(s->data);
151                 flag = true;
152                 return;
153             } s = s->link;
154         } while (s != NULL);
155         if (flag == false) cout << "\nMa sieu thi vua nhap khong co trong sieu thi";
156     } else cout << "\n Danh sach rong\n";
157 }
```

Dòng 143 – 156: - Như mọi khi ta đều mang tinh rằng buộc cao kiểm tra List khác NULL or Không để thực hiện lệnh nên bây giờ ta đi vào bên trong hàm cho 1 cờ hiệu bắt đầu là false để kiểm tra .

- Khi mà mã vừa nhập chạy hết chương trình không thấy sẽ xuất ra Mã Siêu Thị vừa nhập không có trong Siêu Thị ( nếu flag không đổi sang true )

- Lần này ta vẫn dùng duyệt xuất ra từng siêu thị nếu tìm thấy thì đổi flag = true.

### Kết Quả:

```
Nhap lua chon: 3
Nhap ma sieu thi can kiem tra: 3
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
3              3              3              3              3
```

### Kết Quả nếu nhập mã siêu thị không có trong danh sách(tương tự từ câu 1-10):

```
Nhap lua chon: 3
Nhap ma sieu thi can kiem tra: 2
Ma sieu thi vua nhap khong co trong sieu thi
-----MENU-----
```

## Câu 4: In ra màn hình thông tin Siêu thị có doanh số lớn nhất và nhỏ nhất:

### Hàm max doanh số cơ bản:

```
159 //Cau 4
160 void MaxDSCB(qlst s)
161 {
162     if (!IsEmpty(s))
163     {
164         if (s->link != NULL)
165         {
166             qlst node = s; qlst kq = s;
167             int max = s->data.dscb;
168             while (node != NULL)
169             {
170                 if (max < node->data.dscb)
171                 {
172                     max = node->data.dscb;
173                     kq = node;
174                 } node = node->link;
175             } cout << "\nSiêu thị có doanh số cơ bản lớn nhất: ";
176             KhungXuat();
177             Xuat1ST(kq->data);
178         }
179         else
180         {
181             cout << "\nDanh sách chỉ có 1 nên xuất luôn: ";
182             KhungXuat();
183             Xuat1ST(s->data);
184         }
185     } else cout << "\nDanh sách rỗng\n";
186 }
```

Lần này ta xét 3 trường hợp List có NULL, không NULL, có 1 List duy nhất. Nếu có 1 List duy nhất ta xuất luôn data của List (s) ở dòng 179 – 183

Nếu List NULL thì xuất Danh Sách Rỗng.

Còn lại ta thực hiện chương trình như ở dòng 166 - 177:

- Tạo kiểu dữ liệu int max gán bằng doanh số cơ bản siêu thị đầu tiên.
- Cho node và kq gán bằng s dùng node duyệt nếu max < data doanh số cơ bản của node hiện tại.
- Thì max sẽ đổi sang số đó, con trỏ struct kq sẽ chuyển sang node.
- Sau khi chạy hết duyệt thì xuất ra siêu thị chứa kq->data.



## Kết Quả:

```
Nhap lua chon: 4

Sieu thi co doanh so co ban lon nhat:
Ma sieu thi  Ten sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
9              9              9              9              9
```

## Kết quả khi danh sách chỉ có 1 node (tương tự với case 5-7):

```
Nhap lua chon: 4

Danh sach chi co 1 nen xuat lun:
Ma sieu thi  Ten sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
5              5              5              5              5
```

## Hàm Min doanh số cơ bản:

```
188 void MinDSCB(qlst s)
189 {
190     if (!IsEmpty(s))
191     {
192         if (s->link != NULL)
193         {
194             qlst node = s; qlst kq = s;
195             int min = s->data.dscb;
196             while (node != NULL)
197             {
198                 if (min > node->data.dscb)
199                 {
200                     min = node->data.dscb;
201                     kq = node;
202                 } node = node->link;
203             } cout << "\nSieu thi co doanh so co ban nho nhat: ";
204             KhungXuat();
205             Xuat1ST(kq->data);
206         }
207     } else
208     {
209         cout << "\nDanh sach chi co 1 nen xuat lun: ";
210         KhungXuat();
211         Xuat1ST(s->data);
212     }
213     } else cout << "\nDanh sach rong\n";
214 }
```

Lần này ta xét 3 trường hợp List có NULL, không NULL, có 1 List duy nhất.  
Nếu có 1 List duy nhất ta xuất luôn data của List (s) ở dòng 207–211  
Nếu List NULL thì xuất Danh Sách Rỗng.

Còn lại ta thực hiện chương trình như ở dòng 194 -205:

- Tạo kiểu dữ liệu int max gán bằng doanh số cơ bản siêu thị đầu tiên.
- Cho node và kq gán bằng s dùng node duyệt nếu min > data doanh số cơ bản của node hiện tại.
- Thì min sẽ đổi sang số đó, con trỏ struct kq sẽ chuyển sang node.
- Sau khi chạy hết duyệt thì xuất ra siêu thị chứa kq->data.

## Kết Quả:

```
Nhap lua chon: 5

Sieu thi co doanh so co ban nho nhat:
Ma sieu thi  Ten sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
3              3              3              3              3
```

**Câu 5: In ra màn hình thông tin Siêu thị có số lượng nhân viên lớn nhất và nhỏ nhất:**

## Hàm max số lượng nhân viên:

```
216 //Cau 5
217 void MaxSLNV(qlst s)
218 {
219     if (!IsEmpty(s))
220     {
221         if (s->link != NULL)
222         {
223             qlst node = s; qlst kq = s;
224             int max = s->data.dscb;
225             while (node != NULL)
226             {
227                 if (max < node->data.slnv)
228                 {
229                     max = node->data.slnv;
230                     kq = node;
231                 }
232                 node = node->link;
233             }
234             cout << "\nSieu thi co doanh so co ban lon nhat: ";
235             KhungXuat();
236             Xuat1ST(kq->data);
237         }
238         else
239         {
240             cout << "\nDanh sach chi co 1 nen xuat luon: ";
241             KhungXuat();
242             Xuat1ST(s->data);
243         }
244     }
245     else cout << "\nDanh sach rong\n";
246 }
```

Lần này ta xét 3 trường hợp List có NULL, không NULL, có 1 List duy nhất.  
 Nếu có 1 List duy nhất ta xuất luôn data của List (s) ở dòng 236 – 240  
 Nếu List NULL thì xuất Danh Sách Rỗng.

Còn lại ta thực hiện chương trình như ở dòng 223 – 234 :

- Tạo kiểu dữ liệu int max gán bằng số lượng nhân viên siêu thị đầu tiên.
- Cho node và kq gán bằng s dùng node duyệt nếu max < data số lượng nhân viên của node hiện tại.
- Thì max sẽ đổi sang số đó, con trỏ struct kq sẽ chuyển sang node.
- Sau khi chạy hết duyệt thì xuất ra siêu thị chứa kq->data.

## Kết Quả:

```
Nhap lua chon:6
Sieu thi co doanh so co ban lon nhat:
Ma sieu thi  Ten sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
9              9              9              9              9
```

## Hàm min số lượng nhân viên:

```
245 void MinSLNV(qlst s)
246 {
247     if (!IsEmpty(s))
248     {
249         if (s->link != NULL)
250         {
251             qlst node = s; qlst kq = s;
252             int min = s->data.slnv;
253             while (node != NULL)
254             {
255                 if (min > node->data.slnv)
256                 {
257                     min = node->data.slnv;
258                     kq = node;
259                 } node = node->link;
260             } cout << "\nSieu thi co doanh so co ban nho nhat: ";
261             KhungXuat();
262             Xuat1ST(kq->data);
263         }
264         else
265         {
266             cout << "\nDanh sach chi co 1 nen xuat luon: ";
267             KhungXuat();
268             Xuat1ST(s->data);
269         }
270     } else cout << "\nDanh sach rong\n";
271 }
```

Lần này ta xét 3 trường hợp List có NULL, không NULL, có 1 List duy nhất.  
Nếu có 1 List duy nhất ta xuất luôn data của List (s) ở dòng 264 – 268  
Nếu List NULL thì xuất Danh Sách Rỗng.

Còn lại ta thực hiện chương trình như ở dòng 250 -262:

- Tạo kiểu dữ liệu int max gán bằng doanh số cơ bản siêu thị đầu tiên.
- Cho node và kq gán bằng s dùng node duyệt nếu min > data doanh số cơ bản của node hiện tại.
- Thì min sẽ đổi sang số đó, con trỏ struct kq sẽ chuyển sang node.
- Sau khi chạy hết duyệt thì xuất ra siêu thị chứa kq->data.

### Kết Quả:

```
Nhap lua chon: 7

Siêu thị cơ doanh số cơ bản nhỏ nhất:
Ma siêu thịTensuper thị      Nam thanh lap      Doanh thu cơ bản      Số lương nhân viên
3                3                3                3                3
```

### Câu 6: Xuất ra File Text thông kê số lượng nhân viên dưới 100, trên 200 và 100-200:

Hàm kiểu int xuất file thống kê số lượng nhân viên:

```
273 //Cau 6
274 int Xuat_FILE_TK_SLNV(qlst s)
275 {
276     ofstream File;
277     File.open("C:\\Users\\ACER\\OneDrive\\Máy tính\\output.txt", ios_base :: out);
278     if (File.fail()) return 0;
279     int dem100 = 0, dem200 = 0, dem = 0;
280     while (s!=NULL)
281     {
282         if (s->data.dscb < 100) dem100++;
283         else if(s->data.dscb <= 200) dem++;
284         else dem200++;
285         s = s->link;
286     }
287     File << "\n===== Xuat File thong ke so luong nhan vien =====";
288     File << "\nDoanh so co ban duoi 100 la: " << dem100;
289     File << "\nDoanh so co ban 100-200 la: " << dem;
290     File << "\nDoanh so co ban tren 200 la: " << dem200;
291     File << "\n===== Ket Thuc =====";
292     File.close();
293 }
```

Dòng 276: gọi ofstream cung cấp chức năng luồng file đầu ra đặt tên là File.

Dòng 277: dùng lệnh open, ios\_base::out để mở file với đường dẫn như trên hình xà xuất.

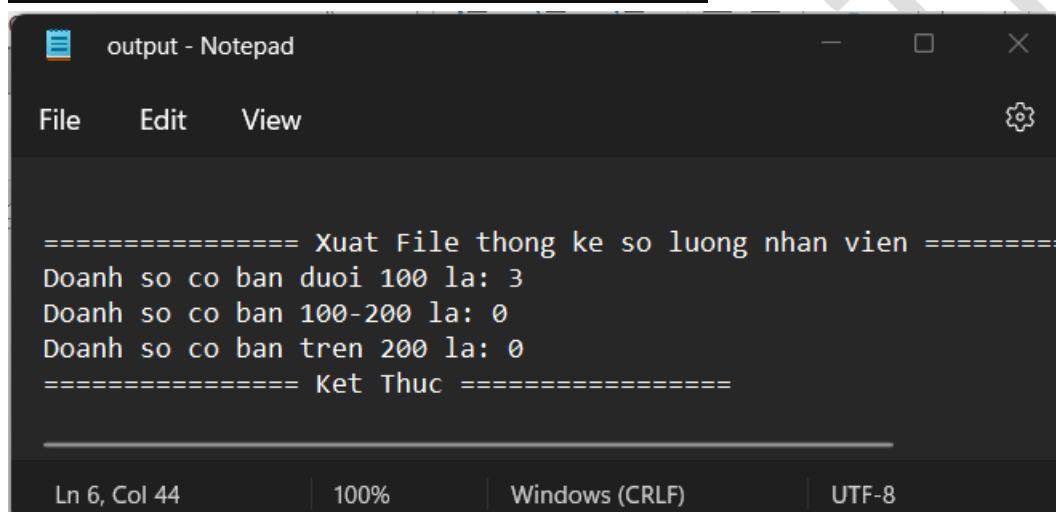
Dòng 278: nếu gọi file thất bại thì return 0.

Dòng 280 – 285: Duyệt s kiểm tra doanh số cơ bản có thỏa từng điều kiện mà đề cho không, rồi tăng các biến đếm mà ta đã khai báo ở dòng 279.

Dòng 287 – 292: Xuất thông tin đã thống kê số lượng nhân viên xong.

Kết quả:

```
Nhap lua chon: 8
Xuat File thong ke so luong nhan vien thanh cong
```



```
output - Notepad
File Edit View

===== Xuat File thong ke so luong nhan vien =====
Doanh so co ban duoi 100 la: 3
Doanh so co ban 100-200 la: 0
Doanh so co ban tren 200 la: 0
===== Ket Thuc =====

Ln 6, Col 44    100%    Windows (CRLF)    UTF-8
```

**Câu 7: Tính doanh số của siêu thị (DS) từ năm thành lập đến bây giờ (Year):**

**Hàm tính doanh số:**

```
313 }
314 }
315 }
316 }
317 }
318 }
319 }
320 }
321 }
322 }
323 }
324 }
325 }
326 }
327 }
328 }
329 }
330 }
331 }
332 }
333 }
334 }
335 }
336 }
337 }
338 }
339 }
340 }
341 }
342 }
343 }
344 }
345 }
346 }
347 }
348 }
349 }
350 }
351 }
352 }
353 }
354 }
355 }
356 }
357 }
358 }
359 }
360 }
361 }
362 }
363 }
364 }
365 }
366 }
367 }
368 }
369 }
370 }
371 }
372 }
373 }
374 }
375 }
376 }
377 }
378 }
379 }
380 }
381 }
382 }
383 }
384 }
385 }
386 }
387 }
388 }
389 }
390 }
391 }
392 }
393 }
394 }
395 }
396 }
397 }
398 }
399 }
400 }
401 }
402 }
403 }
404 }
405 }
406 }
407 }
408 }
409 }
410 }
411 }
412 }
413 }
414 }
415 }
416 }
417 }
418 }
419 }
420 }
421 }
422 }
423 }
424 }
425 }
426 }
427 }
428 }
429 }
430 }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }
454 }
455 }
456 }
457 }
458 }
459 }
460 }
461 }
462 }
463 }
464 }
465 }
466 }
467 }
468 }
469 }
470 }
471 }
472 }
473 }
474 }
475 }
476 }
477 }
478 }
479 }
480 }
481 }
482 }
483 }
484 }
485 }
486 }
487 }
488 }
489 }
490 }
491 }
492 }
493 }
494 }
495 }
496 }
497 }
498 }
499 }
500 }
```

Dòng 298 – 311:

- Vẫn dùng 2 ràng buộc về List .
- Sau đó thực hiện khai báo type time cập nhật thời gian hiện tại để lấy năm hiện tại.
- Ý tưởng bài này dựa vào thực tế tìm siêu thị cần tính doanh số = mã siêu thị.
- Duyệt nếu tìm ra được thì áp dụng công thức như đề cho sau đó dừng vòng lặp.

**Kết Quả nếu nhập mã không có trong danh sách thì dừng:**

```
Nhap lua chon: 9
Nhap ma sieu thi can tim. De tinh doanh so: 2
```

**Kết Quả sau khi thỏa các điều kiện trong chương trình:**

```
Nhap lua chon: 9
Nhap ma sieu thi can tim. De tinh doanh so: 3
Doanh so can tinh: 6057
```

**Câu 8: Nhập lương cơ bản của nhân viên trong 1 năm (LCB). Tính lợi nhuận của siêu thị trong 1 năm (LN):**

**Hàm tính lợi nhuận:**

```
315 int LoiNhanu(qlst s, int lcb, char mstds[50])
316 {
317     if (!IsEmpty(s))
318     {
319         time_t t = time(0);
320         qlst node = s;
321         do
322         {
323             if (strcmp(s->data.mst, mstds) == 0)
324             {
325                 return (s->data.dscb * ((localtime(&t)->tm_year + 1900) - s->data.ntl)) - ( lcb * s->data.slnv);
326             } s = s->link;
327         } while (s != NULL);
328     }
329     else return -1;
330 }
```

Dòng 317- 329:

- Vẫn dùng 2 ràng buộc về List ( List == NULL thì return -1 ) còn không thì chạy chương trình.

- Sau đó thực hiện khai báo type time cập nhật thời gian hiện tại để lấy năm hiện tại.
- Duyệt nếu tìm ra được thì áp dụng công thức như đề cho sau đó return.

### Kết Quả khi nhập mã không có trong danh sách:

```
Nhap lua chon: 10
Nhap maieu thi can tim. De tinh loi nhuan: 2
Nhap lai maieu thi, khong tim thay:
```

### Kết quả khi thỏa mãn, giả sử lương cơ bản là 1000:

```
Nhap lai maieu thi, khong tim thay: 3
Nhap luong co ban cua nhan vien trong 1 nam: 1000
Loi nhuan la: 3057
```

**Câu 9: Nhập vào 1 con số lợi nhuận (SLN) và liệt kê các siêu thị có lợi nhuận > SLN:**

### Hàm đếm:

```
332 //Cau 9
333 int CountNode(qlst s)
334 {
335     int dem = 0;
336     do
337     {
338         dem++;
339         s = s->link;
340     } while (s != NULL);
341     return dem;
342 }
```

## Hàm liệt kê siêu thị có lợi nhuận lớn hơn:

```
344 void ListsWithGreaterProfits(qlst s)
345 {
346     int sln, lcb;
347     if (!IsEmpty(s))
348     {
349         qlst node = s; qlst ln= s;
350         int i = 1, a[10], z = 1, j = 1;
351         do
352         {
353             do
354             {
355                 cout << "\nNhap luong co ban nhan vien thu (" << i << "), de the vao cong thuc: ";
356                 cin >> lcb;
357                 a[i] = lcb;
358                 if (lcb < 0)
359                 {
360                     cout << "\nSo luong co ban, ban nhap duoi am ";
361                     cout << "\nXin vui long nhap lai";
362                 } while (lcb < 0);
363             } while (lcb < 0);
364             i++;
365         } while (i <= CountNode(s));
366         do
367         {
368             cout << "\nLoi nhuan sieu thi thu (" << z << "): "<< LoiNhuhan(ln, a[z],ln->data.mst);
369             z++;
370             ln = ln->link;
371         } while (ln!=NULL);
372         do
373         {
374             cout << "\n Nhap vao 1 con so loi nhuan: ";
375             cin >> sln;
376             if (sln < 0)
377             {
378                 cout << "\nSo loi nhuan, ban nhap duoi am ";
379                 cout << "\nXin vui long nhap lai";
380             } while (sln < 0);
381             cout << "\n===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====\n";
382             KhungXuat();
383             bool flag = false;
384             do
385             {
386                 if (sln < LoiNhuhan(node, a[j], node->data.mst))
387                 {
388                     Xuat1ST(node->data);
389                     flag = true;
390                     j++;
391                     node = node->link;
392                 } while (node != NULL);
393             } while (node != NULL);
394             cout << "\n===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====\n";
395             //cho co flag de check xem duyet toi NULL ko tim thay sln nhap lon hon moi loi nhuan tat ca sieu thi
396             if (flag == false) cout << "\nSo loi nhuan vua nhap la lon nhat nen ko in ra duoc danh sach\n";
397             }else cout << "\nDanh sach sieu thi rong\n";
398         }
```

Vẫn lấy 2 tính ràng buộc của List xong thực hiện chương trình.

Tạo kiểu dữ liệu int: sln (số lợi nhuận) lcb (lương cơ bản).

Tạo con trỏ struct node và ln.

Ý tưởng làm bài này là nhập lương cơ bản nhân viên trong 1 năm từng siêu thị (dòng 351-365)



Sau đó xuất ra lợi nhuận từng siêu thị (dòng 366-371) tiếp theo ta mới ghi số lợi nhuận cần so sánh.

Ta cho cờ hiệu flag bắt đầu false nhằm mục đích nếu so sánh không thấy lợi nhuận nào lớn hơn suy ra lợi nhuận ta nhập là lớn nhất.

### Kết quả, Giả sử lương cơ bản mỗi nhân viên là 1000:

```
Nhap lua chon: 11
Nhap luong co ban nhan vien thu (1), de the vao cong thuc: 1000
Nhap luong co ban nhan vien thu (2), de the vao cong thuc: 1000
Nhap luong co ban nhan vien thu (3), de the vao cong thuc: 1000
Loi nhuan sieu thi thu (1): 9117
Loi nhuan sieu thi thu (2): 3057
Loi nhuan sieu thi thu (3): 5085
Nhap vao 1 con so loi nhuan: 5000

===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien
9              9              9              9              9
5              5              5              5              5
===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====
```

### Kết quả nếu số lợi nhuận nhập vào lớn hơn tất cả:

```
Nhap lua chon: 11
Nhap luong co ban nhan vien thu (1), de the vao cong thuc: 1000
Nhap luong co ban nhan vien thu (2), de the vao cong thuc: 1000
Nhap luong co ban nhan vien thu (3), de the vao cong thuc: 1000
Loi nhuan sieu thi thu (1): 9117
Loi nhuan sieu thi thu (2): 3057
Loi nhuan sieu thi thu (3): 5085
Nhap vao 1 con so loi nhuan: 10000

===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban      So luong nhan vien

===== Danh sach sieu thi co loi nhuan lon hon so loi nhuan da nhap =====
So loi nhuan vua nhap la lon nhat nen ko in ra duoc danh sach
```

## Câu 10: Sắp xếp danh sách tăng dần theo năm thành lập:

### Hàm swap:

```
400 //Cau 10
401 void Swap(qlst a, qlst b)
402 {
403     st temp = a->data;
404     a->data = b->data;
405     b->data = temp;
406 }
```

### Hàm Selection Sort tăng dần:

```
408 void DSSTangDan(qlst& s)
409 {
410     time_t t = time(0);
411     if (!IsEmpty(s))
412     {
413         qlst node = s;
414         qlst after = s->link;
415         qlst min = s;
416         cout << "\n===== Danh sach da sap xep tang dan =====";
417         if (s->link != NULL)
418         {
419             do
420             {
421                 min = node;
422                 after = node->link;
423                 do
424                 {
425                     if (((localtime(&t)->tm_year + 1900) - after->data.ntl) < ((localtime(&t)->tm_year + 1900) - min->data.ntl))
426                         min = after;
427                     after = after->link;
428                 } while (after != NULL);
429                 if (strcmp(min->data.mst, node->data.mst) != 0)
430                     Swap(node, min);
431                 node = node->link;
432             } while (node->link != NULL);
433             XuatDSST(s);
434         }
435     }
436     else
437     {
438         cout << "\nDanh sach chi co 1 nen: ";
439         KhungXuat();
440         Xuat1ST(s->data);
441     }
442     cout << "\n===== Danh sach da sap xep tang dan =====\n";
443     } else cout << "\nDanh sach rong\n";
}
```

Cho tạo con trỏ struct gồm node, min và after thì gán tại vị trí node sau First List (s).

Thực hiện 3 ràng buộc của List: NULL, Không NULL, 1 List duy nhất.

Nếu 1 List duy nhất thì xuất ra luôn ở dòng 435-440.

Cho min gán tại vị trí after trước mỗi lần lặp after cũng sẽ cập nhật tại vị trí sau con trỏ struct node.

Sau khi so sánh thỏa điều kiện của đề ta bắt đầu gán con trỏ struct min tại after để nhằm mục đích sau khi after chạy hết vòng lặp.

Lấy mã siêu thị tại struct min so sánh với node mà khác thì ta tiến hành swap data của min và node cứ thế mà thực hiện hết vòng duyệt node.

### Kết quả:

```
Nhap lua chon: 12
===== Danh sach da sap xep tang dan =====
Danh sach sieu thi
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban    So luong nhan vien
9          9                9                9                9
5          5                5                5                5
3          3                3                3                3
===== Danh sach da sap xep tang dan =====
```

### Kết quả khi danh sách có 1 node:

```
Nhap lua chon: 12
===== Danh sach da sap xep tang dan =====
Danh sach chi co 1 nen:
Ma sieu thiTen sieu thi      Nam thanh lap      Doanh thu co ban    So luong nhan vien
5          5                5                5                5
===== Danh sach da sap xep tang dan =====
```

## Hàm Menu:

```
466 void DSMenu(qlst s)
467 {
468     KhoiTao(s);
469     do {
470         int choose;
471         Menu();
472         cout << "\nNhap lua chon: ";
473         cin >> choose;
474         switch (choose)
475         {
476             case 1:
477                 NhapDSST(s);
478                 break;
479             case 2:
480                 XuatDSST(s);
481                 break;
482             case 3:
483                 char mstkt[50];
484                 cout << "\nNhap ma sieu thi can kiem tra: ";
485                 cin >> mstkt;
486                 KT_MST(s, mstkt);
487                 break;
488             case 4:
489                 MaxDSCB(s);
490                 break;
491             case 5:
492                 MinDSCB(s);
493                 break;
494             case 6:
495                 MaxSLNV(s);
496                 break;
```

```

508         case 9:
509             char mstds[50];
510             cout << "\nNhap ma sieu thi can tim. De tinh doanh so: ";
511             cin >> mstds;
512             while (!KT_TrungMa(s, mstds))
513             {
514                 cout << "\nNhap lai ma sieu thi, khong tim thay: ";
515                 cin >> mstds;
516             }
517             DoanhSo(s, mstds);
518             break;
519         case 10:
520             int loinhuan;
521             int lcb;
522             char mstln[50];
523             cout << "\nNhap ma sieu thi can tim. De tinh loi nhuan: ";
524             cin >> mstln;
525             while (!KT_TrungMa(s, mstln))
526             {
527                 cout << "\nNhap lai ma sieu thi, khong tim thay: ";
528                 cin >> mstln;
529             }

```

```

530     do {
531         cout << "\nNhap luong co ban cua nhan vien trong 1 nam: ";
532         cin >> lcb;
533         if (lcb < 0)
534         {
535             cout << "\nSo luong co ban, ban nhap duoi am ";
536             cout << "\nXin vui long nhap lai";
537         }
538     } while (lcb < 0);
539     loinhuan = LoiNhuan(s, lcb, mstln);
540     if (loinhuan == -1) cout << "\nMa sieu thi can tim ko ton tai or Danh sach sieu thi rong";
541     else
542         cout << "\nLoi nhuan la: " << loinhuan << endl;
543     break;
544     case 11:
545         ListsWithGreaterProfits(s);
546         break;
547     case 12:
548         DSSTangDan(s);
549         break;
550     case 0:
551         cout << "\nCAM ON CO DA HUONG DAN EM LAM DO AN TIN HOC";
552         cout << "\nKET THUC CHUONG TRINH. THOAT!!!\n";
553         system("pause");
554         GiaiPhong(s);
555         exit(1);
556         break;
557     default:
558         cout << "\nNHAP NGOAI PHAM VI MENU XINH VUI LONG NHAP LAI\n";
559         break;
560     }
561 } while (1);

```

Ta check xem mã nhập có trong danh sách không bằng cách lặp lại nếu bool false của hàm KT\_TrungMa.(dòng: 512-516 và 525-529).

## Kết quả khi ta nhập ngoại phạm vi menu và thoát:

```
Nhap lua chon: 13
```

```
NHAP NGOAI PHAM VI MENU XINH VUI LONG NHAP LAI
```

```
Nhap lua chon: 0
```

```
CAM ON CO DA HUONG DAN EM LAM DO AN TIN HOC  
KET THUC CHUONG TRINH. THOAT!!!  
Press any key to continue . . .
```

```
D:\NguyenBuiNhuty\x64\Debug\NguyenBuiNhuty.exe (process 8424) exited with code 1.  
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.  
Press any key to close this window . . .
```

