**Nộp bài: 2 file trong đó 1 file word và 1 file cpp**

**Các em in nộp quyển trong đó:**

**1/ Phần code**

**2/ Phần kết quả chạy chương trình của từng câu.**

* **Tờ bìa file word ghi rõ họ và tên, xuất học theo thời khóa biểu.**
* **Không in màu và không đóng bìa gương.**

**Thời gian nộp: đến 18h ngày 24/12/2023 trên ms team trong thư mục bài cuối kỳ.**

**Đề 1:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tính kích thước của cây (số nút)
4. Sao chép cây
5. Xóa cây

**Đề 2:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách tăng dần theo mã số sinh viên tăng dần.

3. Tìm vị trí node của sinh viên có năm sinh lớn nhất.  
4. Sắp xếp danh sách tăng dần theo tên (thứ tự bảng chữ cái) và năm sinh (nếu trùng tên thì sắp theo năm sinh tăng dần).

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tìm min
4. Tìm min của cây con bên phải
5. Xóa cây

**Đề 3:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách giảm dần theo năm sinh.

3. Tìm vị trí node của sinh viên có mã sinh viên lớn nhất.

4. Đếm xem có bao nhiêu sinh viên sinh năm 1990.

5. Xóa hết danh sách trên.

**Bài 2:**Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 4:**

**Bài 1:** Cho dãy số theo thứ tự nhập từ trái sang phải như sau:

**20 15 35 30 11 13 17 36 47 16 38 28 14**

1. Hãy vẽ cây nhị phân tìm kiếm cho dãy số trên.
2. Hãy cho biết kết quả duyệt cây trên theo thứ tự trước, giữa và sau.
3. Cho biết độ cao của cây, các nút lá, các nút có bậc 2.
4. Vẽ lại cây sau khi thêm nút: 25 và 91

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** với các thuật toán sắp xếp đã học: *Selection Sort*, *Quick sort*, *Merge sort, Shell sort* và *Heap sort*. Trong đó, yêu cầu cụ thể như sau:

a) Selection Sort: Số âm tăng dần về cuối dãy, số không âm giảm dần về đầu dãy.

c) Merge Sort: Trộn theo cả 2 cách (tự nhiên và trực tiếp). Sắp tăng dần.

d) Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).

**Nộp bài: 2 file trong đó 1 file word và 1 file cpp**

**Các em in nộp quyển trong đó:**

**1/ Phần code**

**2/ Phần kết quả chạy chương trình của từng câu.**

* **Tờ bìa file word ghi rõ họ và tên, xuất học theo thời khóa biểu.**
* **Không in màu và không đóng bìa gương.**

**Thời gian nộp: đến 18h ngày 24/12/2023 trên ms team trong thư mục bài cuối kỳ.**

**Đề 5:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** yêu cầu cụ thể như sau:

1. Quick Sort: Sắp tăng dần theo trị tuyệt đối.
2. Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).
3. Heap Sort: Sắp giảm dần

**Đề 6:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh viên thuộc lớp “22T3”.
3. Sắp sếp danh sách sinh viên theo điểm cuối kỳ tăng dần
4. Xóa phần tử đầu khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây  
2. Đếm số nút lá trong cây  
3. Tính kích thước của cây (số nút)

4. Tìm nút nhỏ nhất của cây nhị phân.

**Đề 7:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh có tên là “Hoa”.
3. Sắp sếp danh sách sinh viên theo tên sinh viên tăng dần
4. Xóa phần tử cuối khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

***Bài 2:*** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** hãy sắp xếp dãy số trên theo*Merge sort, Shell sort* và *Heap sort*.

**Đề 8:**

**Bài 1:** Viết chương trình quản lý NHÂN SỰ của một công ty bằng danh sách liên kết đơn với  
thông tin mỗi NHÂN VIÊN gồm: Mã nhân viên, Tên nhân viên, Giới tính, Năm sinh, Chức vụ, Lương.  
  
Viết chương trình các chức năng thực hiện như sau:  
1. Nhập danh sách gồm n nhân viên với đầy đủ các thông tin.  
2. Nhập lương cơ bản và tính lương nhân viên bằng **lương cơ bản** \* **chức vụ**.  
3. Nhập vào 1 **mã nhân viên** và kiểm tra mã nhân viên đó có tồn tại hay không?  
Nếu có xuất ra màn hình thông tin của nhân viên đó.  
4. Sắp xếp danh sách tăng dần theo **năm sinh**.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 9:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tính kích thước của cây (số nút)
4. Sao chép cây
5. Xóa cây

**Đề 10:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách tăng dần theo mã số sinh viên tăng dần.

3. Tìm vị trí node của sinh viên có năm sinh lớn nhất.  
4. Sắp xếp danh sách tăng dần theo tên (thứ tự bảng chữ cái) và năm sinh (nếu trùng tên thì sắp theo năm sinh tăng dần).

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tìm min
4. Tìm min của cây con bên phải
5. Xóa cây

**Đề 11:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách giảm dần theo năm sinh.

3. Tìm vị trí node của sinh viên có mã sinh viên lớn nhất.

4. Đếm xem có bao nhiêu sinh viên sinh năm 1990.

5. Xóa hết danh sách trên.

**Bài 2:**Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 12:**

**Bài 1:** Cho dãy số theo thứ tự nhập từ trái sang phải như sau:

**20 15 35 30 11 13 17 36 47 16 38 28 14**

1. Hãy vẽ cây nhị phân tìm kiếm cho dãy số trên.
2. Hãy cho biết kết quả duyệt cây trên theo thứ tự trước, giữa và sau.
3. Cho biết độ cao của cây, các nút lá, các nút có bậc 2.
4. Vẽ lại cây sau khi thêm nút: 25 và 91

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** với các thuật toán sắp xếp đã học: *Selection Sort*, *Quick sort*, *Merge sort, Shell sort* và *Heap sort*. Trong đó, yêu cầu cụ thể như sau:

a) Selection Sort: Số âm tăng dần về cuối dãy, số không âm giảm dần về đầu dãy.

c) Merge Sort: Trộn theo cả 2 cách (tự nhiên và trực tiếp). Sắp tăng dần.

d) Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).

**Đề 13:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** yêu cầu cụ thể như sau:

1. Quick Sort: Sắp tăng dần theo trị tuyệt đối.
2. Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).
3. Heap Sort: Sắp giảm dần

**Đề 14:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh viên thuộc lớp “22T3”.
3. Sắp sếp danh sách sinh viên theo điểm cuối kỳ tăng dần
4. Xóa phần tử đầu khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây  
2. Đếm số nút lá trong cây  
3. Tính kích thước của cây (số nút)

4. Tìm nút nhỏ nhất của cây nhị phân.

**Đề 15:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh có tên là “Hoa”.
3. Sắp sếp danh sách sinh viên theo tên sinh viên tăng dần
4. Xóa phần tử cuối khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

***Bài 2:*** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** hãy sắp xếp dãy số trên theo*Merge sort, Shell sort* và *Heap sort*.

**Đề 16:**

**Bài 1:** Viết chương trình quản lý NHÂN SỰ của một công ty bằng danh sách liên kết đơn với  
thông tin mỗi NHÂN VIÊN gồm: Mã nhân viên, Tên nhân viên, Giới tính, Năm sinh, Chức vụ, Lương.  
  
Viết chương trình các chức năng thực hiện như sau:  
1. Nhập danh sách gồm n nhân viên với đầy đủ các thông tin.  
2. Nhập lương cơ bản và tính lương nhân viên bằng **lương cơ bản** \* **chức vụ**.  
3. Nhập vào 1 **mã nhân viên** và kiểm tra mã nhân viên đó có tồn tại hay không?  
Nếu có xuất ra màn hình thông tin của nhân viên đó.  
4. Sắp xếp danh sách tăng dần theo **năm sinh**.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 17:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tính kích thước của cây (số nút)
4. Sao chép cây
5. Xóa cây

**Đề 18:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách tăng dần theo mã số sinh viên tăng dần.

3. Tìm vị trí node của sinh viên có năm sinh lớn nhất.  
4. Sắp xếp danh sách tăng dần theo tên (thứ tự bảng chữ cái) và năm sinh (nếu trùng tên thì sắp theo năm sinh tăng dần).

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây
2. Đếm số nút lá trong cây
3. Tìm min
4. Tìm min của cây con bên phải
5. Xóa cây

**Đề 19:**

**Bài 1: Cho một danh sách liên kết đơn gồm các sinh viên sau:**

****

1. Tạo một cấu trúc dữ liệu để xử lý danh sách trên.  
2. Sắp xếp danh sách giảm dần theo năm sinh.

3. Tìm vị trí node của sinh viên có mã sinh viên lớn nhất.

4. Đếm xem có bao nhiêu sinh viên sinh năm 1990.

5. Xóa hết danh sách trên.

**Bài 2:**Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 20:**

**Bài 1:** Cho dãy số theo thứ tự nhập từ trái sang phải như sau:

**20 15 35 30 11 13 17 36 47 16 38 28 14**

1. Hãy vẽ cây nhị phân tìm kiếm cho dãy số trên.
2. Hãy cho biết kết quả duyệt cây trên theo thứ tự trước, giữa và sau.
3. Cho biết độ cao của cây, các nút lá, các nút có bậc 2.
4. Vẽ lại cây sau khi thêm nút: 25 và 91

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** với các thuật toán sắp xếp đã học: *Selection Sort*, *Quick sort*, *Merge sort, Shell sort* và *Heap sort*. Trong đó, yêu cầu cụ thể như sau:

a) Selection Sort: Số âm tăng dần về cuối dãy, số không âm giảm dần về đầu dãy.

c) Merge Sort: Trộn theo cả 2 cách (tự nhiên và trực tiếp). Sắp tăng dần.

d) Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).

**Đề 21:**

***Bài 1:*** Khai báo cấu trúc dữ liệu để tổ chức danh sách liên kết đơn quản lý điểm của sinh viên gồm các thông tin của mỗi sinh viên bao gồm: Mã sinh viên, tên tên sinh viên, điêm chuyên cần, diểm giữa kỳ, điểm cuối kỳ.

1. Thêm ,sửa, xóa theo mã sinh viên ở vị trí bất kỳ;

2. Tính điểm trung bình của mỗi sinh viện như sau điểm chuyên cần hệ số 2, điểm giữa kỳ hệ số 2, điểm cuỗi kỳ hệ số 6 .

3. Tìm vị trí của sinh viên có điểm trung bình lớn nhất.

4. Sắp xếp tang dần danh sách sinh viên theo điểm trung bình.

**Bài 2:** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** yêu cầu cụ thể như sau:

1. Quick Sort: Sắp tăng dần theo trị tuyệt đối.
2. Shell Sort: Sắp tăng dần với bước chạy tùy chọn (nên lấy theo sách).
3. Heap Sort: Sắp giảm dần

**Đề 22:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh viên thuộc lớp “22T3”.
3. Sắp sếp danh sách sinh viên theo điểm cuối kỳ tăng dần
4. Xóa phần tử đầu khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây  
2. Đếm số nút lá trong cây  
3. Tính kích thước của cây (số nút)

4. Tìm nút nhỏ nhất của cây nhị phân.

**Đề 23:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh có tên là “Hoa”.
3. Sắp sếp danh sách sinh viên theo tên sinh viên tăng dần
4. Xóa phần tử cuối khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

***Bài 2:*** Viết chương trình thực hiện sắp xếp một dãy gồm N (N ≤ 100) **số nguyên** hãy sắp xếp dãy số trên theo*Merge sort, Shell sort* và *Heap sort*.

**Đề 24:**

**Bài 1:** Viết chương trình quản lý NHÂN SỰ của một công ty bằng danh sách liên kết đơn với  
thông tin mỗi NHÂN VIÊN gồm: Mã nhân viên, Tên nhân viên, Giới tính, Năm sinh, Chức vụ, Lương.  
  
Viết chương trình các chức năng thực hiện như sau:  
1. Nhập danh sách gồm n nhân viên với đầy đủ các thông tin.  
2. Nhập lương cơ bản và tính lương nhân viên bằng **lương cơ bản** \* **chức vụ**.  
3. Nhập vào 1 **mã nhân viên** và kiểm tra mã nhân viên đó có tồn tại hay không?  
Nếu có xuất ra màn hình thông tin của nhân viên đó.  
4. Sắp xếp danh sách tăng dần theo **năm sinh**.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính kích thước của cây (số nút)
2. Sao chép cây
3. Tìm x
4. Tìm max
5. Tìm max của cây con bên trái

**Đề 25:**

**Bài 1:** Viết  chương trình chính: Sử dụng cấu trúc danh sách liên kết đơn và các thao tác ở trên để:

– Nhập vào một danh sách liên kết đơn gồm n sinh viên (n bất kỳ) gồm các thông tin sau:  Mã Sinh viên (kiểu số nguyên), tên Sinh viên (kiểu chuỗi ký tự), lớp (kiểu chuỗi ký tự), điểm cuối kỳ (kiểu số thực), hạnh kiểm (Tốt, khá, trung bình, yếu).

1. Hiển thị danh sách đã nhập ra màn hình.
2. Liệt kê ra màn hình danh sách tất cả những Sinh viên thuộc lớp “22T3”.
3. Sắp sếp danh sách sinh viên theo điểm cuối kỳ tăng dần
4. Xóa phần tử đầu khỏi danh sách.
5. Hiển thị lại danh sách sau khi xóa ra màn hình.

**Bài 2:** Xây dựng cây nhị phân tìm kiếm để:

1. Tính độ cao của cây  
2. Đếm số nút lá trong cây  
3. Tính kích thước của cây (số nút)

4. Tìm nút nhỏ nhất của cây nhị phân.