**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM KHOA CÔNG NGHỆ THÔNG TIN**

🙞🙞🙜🙜

**MÔN HỌC: CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT**





**GVHD: TS. HUỲNH XUÂN PHỤNG**

**ĐỒ ÁN CUỐI KỲ**

**MÔ PHỎNG DẠNG HÌNH ẢNH ĐỂ BIỂU DIỄN**

**CÂY NHỊ PHÂN DẠNG THREADED VỚI 3 CÁCH DUYỆT CÂY.**

**TÌM KIẾM TRÊN CÂY DẠNG THREADED**

**CHÈN MỘT NODE VÀO CÂY THREADED.**

**Mã môn học: DASA230179\_22\_1\_08**

**NHÓM: 05**

**THÀNH VIÊN:**

Nguyễn Việt Khoa 21110512

Phạm Thị Phương Nghi 21110556

**HỌC KỲ 1 - NĂM HỌC: 2022-2023**

***Nhận xét của giáo viên***

*Ngày 31 tháng 12 năm 2022*

**Giáo viên chấm điểm**

**GV.TS. Huỳnh Xuân Phụ****ng**

**ĐỀ TÀI BỔ SUNG SỐ 3**

-Viết chương trình mô phỏng dạng hình ảnh để biểu diễn cây nhị phân theo dạng threaded với 3 cách duyệt cây.

+ Mô phỏng thuật toán duyệt cây thread.

+ Mô phỏng thuật toán tìm kiếm trên cây dạng threaded.

+ Mô phỏng quá trình chèn một node vào cây dạng threaded.

----------------------------------------------------------------------------------------------------

**BÁO CÁO ĐỒ ÁN**

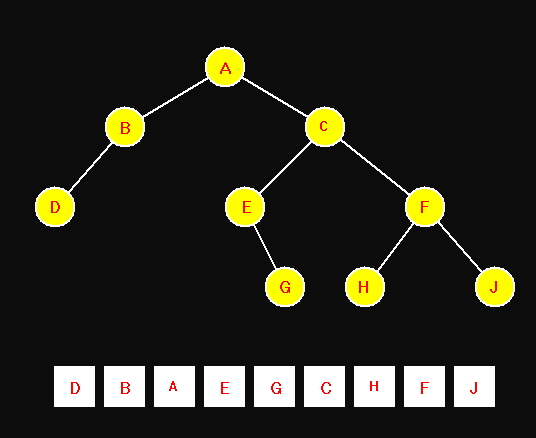
Chương trình đã xây dựng được bao gồm các chức năng:

* Mô phỏng dạng hình ảnh và duyệt cây UnThreaded theo 3 cách:

+ Inorder

Ảnh có chứa văn bản

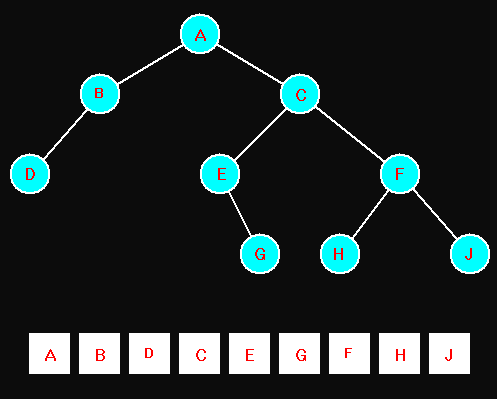
Mô tả được tạo tự động



+ Preorder

Ảnh có chứa văn bản

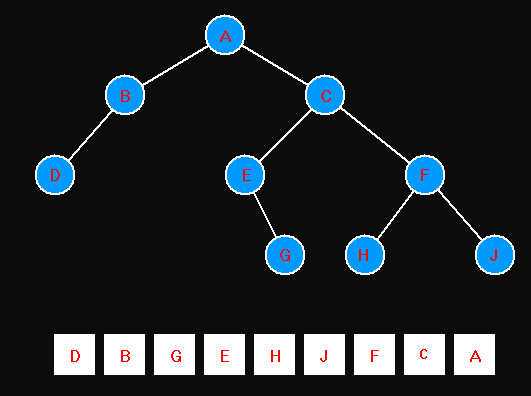
Mô tả được tạo tự động



+ Postorder

Ảnh có chứa văn bản

Mô tả được tạo tự động

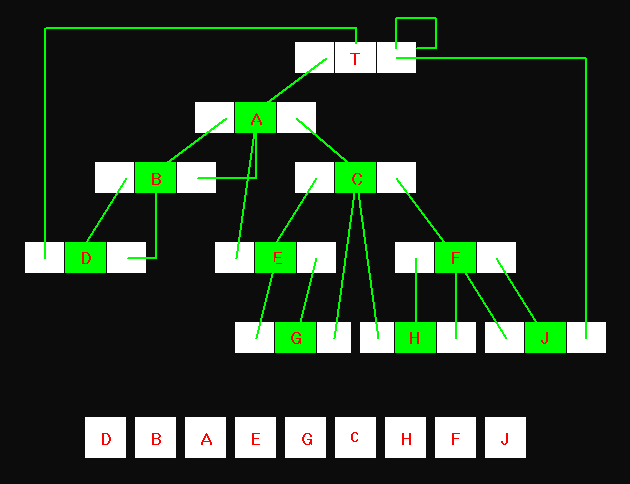


* Mô phỏng dạng hình ảnh và duyệt cây Threaded theo 3 cách:

+ Inorder

Ảnh có chứa văn bản

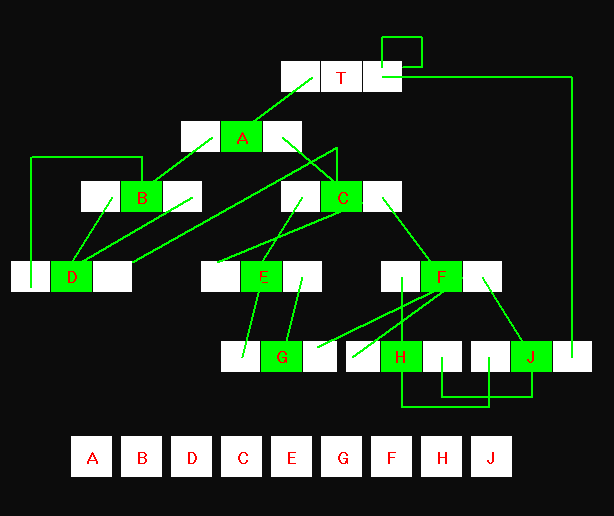
Mô tả được tạo tự động



+ Preorder

Ảnh có chứa văn bản

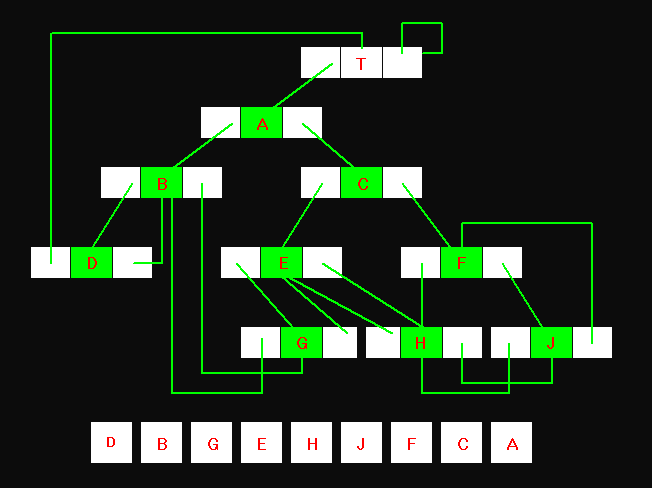
Mô tả được tạo tự động



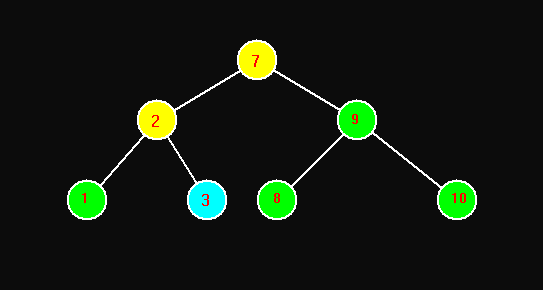
+ Postorder

Ảnh có chứa văn bản

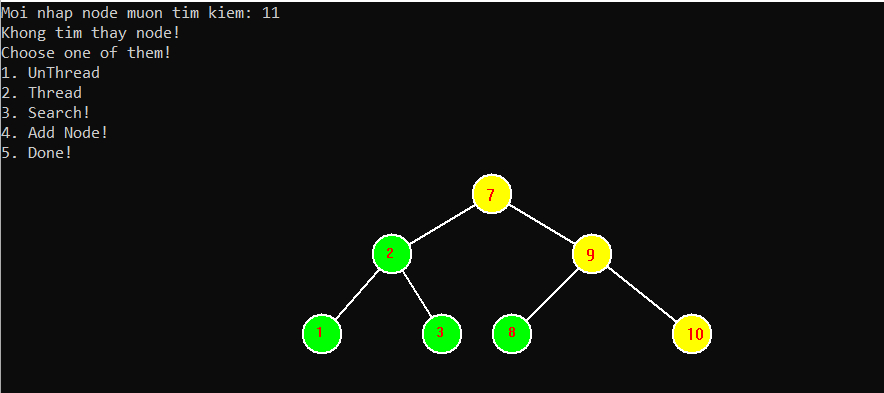
Mô tả được tạo tự động



* Mô phỏng thuật toán tìm kiếm trên cây.

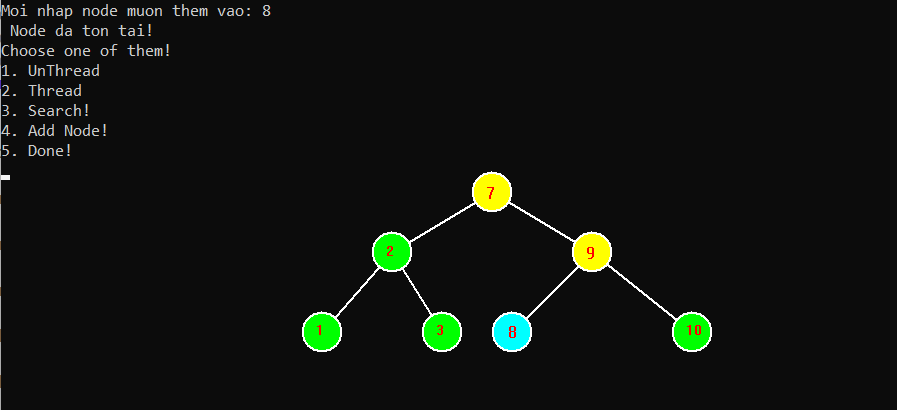


\* Tìm kiếm node 3.

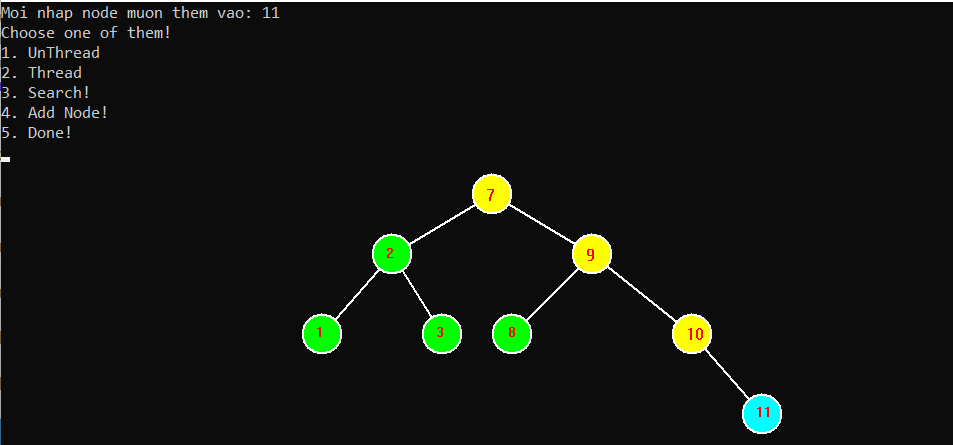


Tìm kiếm node 11 (Không tìm thấy node trong cây).

* Mô phỏng quá trình chèn một node vào cây.



Chèn thêm node 8 (node đã tồn tại).



Chèn thêm node 11 (node chưa có trong cây).

**Chi tiết code đã làm trong đồ án:**

- Dòng 1-8: Khai báo thư viện

- Dòng 11-40: Viết hàm tạo cây dưới dạng chung theo kiểu UnThreaded bằng các hàm line(vẽ đường thẳng), circle(vẽ hình tròn - các node), drawtext (ghi giá trị node) - hàm createTree()

- Dòng 42-110: Xây dựng hàm unThreadInorder để duyệt cây theo kiểu Inorder

+ Dòng 44-53: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 55-108: Duyệt cây theo cách duyệt Inorder (UnThreaded) bằng cách đổi màu các circle (các node) sau đó đưa các node vừa duyệt vào hàng đợi bên dưới

- Dòng 112-180: Xây dựng hàm unThreadPreorder để duyệt cây theo kiểu Preorder

+ Dòng 114-123: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 125-179: Duyệt cây theo cách duyệt Preorder (UnThreaded) bằng cách đổi màu các circle (các node) sau đó đưa các node vừa duyệt vào hàng đợi bên dưới

- Dòng 182-250: Xây dựng hàm unThreadPostorder để duyệt cây theo kiểu Postorder

+ Dòng 184-193: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 195-249: Duyệt cây theo cách duyệt Postorder (UnThreaded) bằng cách đổi màu các circle (các node) sau đó đưa các node vừa duyệt vào hàng đợi bên dưới

- Dòng 252-513: Xây dựng hàm threadInorder để vẽ cây theo kiểu threaded và duyệt cây vừa vẽ theo Inorder. Mỗi node có cấu trúc gồm 3 phần: Left-Info-Right, mỗi phần được biểu diễn bằng một hình chữ nhật (rectangle), khi duyệt cây các phần Left-Info-Right sẽ trỏ tới các node chỉ định theo cách duyệt Inorder (Threaded)

+ Dòng 254-263: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 265-300: vẽ nhánh cây bằng các hàm line (biểu diễn thứ tự trỏ theo cách duyệt Inorder)

+ Dòng 302-351: vẽ các node có cấu trúc gồm 3 phần Left-Info-Right

+ Dòng 353-512: duyệt cây Threaded theo cách duyệt Inorder bằng cách vẽ các node mới khác màu đè lên các node cũ đồng thời thay đổi màu các đường thẳng dùng để trỏ đến các node khác bằng hàm setcolor. Sau khi duyệt xong mỗi node (đổi màu), phần Info của node vừa duyệt sẽ được đưa vào hàng đợi bên dưới.

- Dòng 515-784: Xây dựng hàm threadPreorder để vẽ cây theo kiểu threaded và duyệt cây vừa vẽ theo Preorder. Mỗi node có cấu trúc gồm 3 phần: Left-Info-Right, mỗi phần được biểu diễn bằng một hình chữ nhật (rectangle), khi duyệt cây các phần Left-Info-Right sẽ trỏ tới các node chỉ định theo cách duyệt Preorder (Threaded)

+ Dòng 517-526: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 528-569: vẽ nhánh cây bằng các hàm line (biểu diễn thứ tự trỏ theo cách duyệt Preorder)

+ Dòng 571-620: vẽ các node có cấu trúc gồm 3 phần Left-Info-Right

+ Dòng 622-783: duyệt cây Threaded theo cách duyệt Preorder bằng cách vẽ các node mới khác màu đè lên các node cũ đồng thời thay đổi màu các đường thẳng dùng để trỏ đến các node khác bằng hàm setcolor. Sau khi duyệt xong mỗi node (đổi màu), phần Info của node vừa duyệt sẽ được đưa vào hàng đợi bên dưới.

- Dòng 786-1067: Xây dựng hàm threadPostorder để vẽ cây theo kiểu threaded và duyệt cây vừa vẽ theo Postorder. Mỗi node có cấu trúc gồm 3 phần: Left-Info-Right, mỗi phần được biểu diễn bằng một hình chữ nhật (rectangle), khi duyệt cây các phần Left-Info-Right sẽ trỏ tới các node chỉ định theo cách duyệt Postorder (Threaded)

+ Dòng 788-797: vẽ hàng đợi bằng hàm rectangle (các ô vuông) dùng để chứa các node đã xét.

+ Dòng 799-844: vẽ nhánh cây bằng các hàm line (biểu diễn thứ tự trỏ theo cách duyệt Postorder)

+ Dòng 846-896: vẽ các node có cấu trúc gồm 3 phần Left-Info-Right

+ Dòng 898-1066: duyệt cây Threaded theo cách duyệt Postorder bằng cách vẽ các node mới khác màu đè lên các node cũ đồng thời thay đổi màu các đường thẳng dùng để trỏ đến các node khác bằng hàm setcolor. Sau khi duyệt xong mỗi node (đổi màu), phần Info của node vừa duyệt sẽ được đưa vào hàng đợi bên dưới.

- Dòng 1069-1231: Tạo cây mới trong hàm searchTree và mô phỏng hình ảnh cách tìm kiếm trên cây bằng cách nhập vào phần tử cần tìm, sau đó so sánh phần tử vừa nhập với phần tử đang xét (phần tử xét đầu tiên là node root). Nếu phần tử nhập bằng phần tử đang xét thì thông báo đã tìm thấy node cần tìm và kết thúc quá trình tìm kiếm; nếu phần tử vừa nhập lớn hơn phần tử đang xét thì cây sẽ xét tiếp về phía bên phải; nếu phần tử vừa nhập bé hơn phần tử đang xét thì cây sẽ xét tiếp về phía bên trái. Cứ tiếp tục xét như thế cho đến khi đến node lá, nếu đến cuối cùng vẫn không tìm thấy node nào trùng với node đã nhập thì in ra màn hình “Không tìm thấy node!” và kết thúc quá trình tìm kiếm.

+ Dòng 1071-1076: vẽ các đường thẳng (line) để liên kết các node trong cây

+ Dòng 1078-1091: vẽ các node trong cây bằng hàm circle và drawtext

+ Dòng 1093-1094: chú thích về màu node đã xét và cách tìm kiếm trên cây

+ Dòng 1096-1097: Nhập node muốn tìm kiếm trong cây

+ Dòng 1098-1228: duyệt cây theo cách duyệt đã nêu trên, nếu tìm thấy node cần tìm sẽ đổi màu node đó và kết thúc quá trình tìm, nếu không tìm thấy sẽ in ra màn hình “Không tìm thấy node!”.

- Dòng 1233-1425: Xây dựng hàm addTree, thêm node vào cây hoạt động theo cấu trúc: nhập phần tử cần thêm vào cây, sau đó so sánh phần tử vừa nhập với phần tử đang xét (phần tử xét đầu tiên là node root). Nếu phần tử nhập bằng phần tử đang xét thì thông báo node đã tồn tại và kết thúc quá trình thêm; nếu phần tử vừa nhập lớn hơn phần tử đang xét thì cây sẽ xét tiếp về phía bên phải; nếu phần tử vừa nhập bé hơn phần tử đang xét thì cây sẽ xét tiếp về phía bên trái. Cứ tiếp tục xét như thế cho đến khi đến node lá, nếu đến cuối cùng vẫn không tìm thấy node nào trùng với node đã nhập thì tiến hành thêm node vừa nhập ngay sau node lá vừa kết thúc (thêm bên phải nếu node vừa nhập lớn hơn node lá; ngược lại, thêm bên trái nếu node vừa nhập bé hơn node lá) sau đó kết thúc quá trình thêm.

+ Dòng 1235-1241: vẽ các đường thẳng (line) để liên kết các node trong cây

+ Dòng 1243-1257: vẽ các node trong cây bằng hàm circle và drawtext

+ Dòng 1258-1260: chú thích về màu node đã xét và cách thêm node trên cây

+ Dòng 1262-1263: Nhập node muốn thêm vào cây

+ Dòng 1265-1424: thêm node như cấu trúc đã nêu trong hàm addTree

- Dòng 1428-1527: hàm main bào gồm

+ Dòng 1430: khai báo biến x (nhập trong phần searchTree), y (nhập trong phần addTree)

+ Dòng 1431:

+ Dòng 1432: gọi lại hàm createTree

+ Dòng 1434-1524: tạo ra menu để người dùng lựa chọn gồm 4 lựa chọn:

1. UnThread: trong UnThread lại có thêm 4 lựa chọn lần lượt là duyệt cây theo Inorder, Preorder, Postorder và cuối cùng là trở lại menu
2. Thread: trong Thread lại có thêm 4 lựa chọn lần lượt là duyệt cây theo Inorder, Preorder, Postorder và cuối cùng là trở lại menu
3. Search!: dùng để tìm kiếm node trong cây (searchTree)
4. Add Node!: dùng để thêm node vào cây (addTree)
5. Done!: dùng để kết thúc chương trình

\* Trong quá trình duyệt cây có sử dụng hàm Sleep để nghỉ giữa các thao tác duyệt giúp dễ nhìn hơn