|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Quản lý dự án (chương 3)**   * **Nguyên tắc cơ bản của quản lý dự án** * Giai đoạn khởi đầu dự án,Việc xác định tính khả thi dự án,Hoạt động lập kế hoạch và quản lí,Lập kế hoạch dự án,Quản lí các thành phần phần và phân tích hệ thống. * **Khởi đầu dự án** (Các vấn đề trong tổ chức sẽ dẫn người quản trị tới việc cần tìm ra giải pháp cho vấn đề đó trong hệ thống - Cơ hội nâng cấp. Các vấn đề xảy ra trong hệ thống sẽ là cơ hội để hệ thống được nâng cấp, thay đổi hoặc cài đặt hệ thống mới.) * **Xác định vấn đề**. (Vấn đề này thực sự là cơ hội hay là vấn đề cần giải quyết. - Các vấn đề độc lập giữa thách thức hoạc cơ hội. - Mục tiêu phải phù hợp với vấn đề từng điểm một. - Yêu cầu Những điều cần phải hoàn thành cùng với các giải pháp và hạn chế có thể làm hạn chế sự phát triển của hệ thống.) * **Lựa chọn dự án** (Hỗ trợ từ quản lí - Cam kết thời giản thích hợp của dự án - Khả năng hoàn thiện được mục tiêu của dự án - Thực tế về nguồn lực cho nhà phân tích hệ thống và tổ chức. - Cho thấy sự đáng giá đáng đầu tư hơn các dự án khác.) * **Xác định tính khả thi** (kinh tế, vận hành, kỹ thuật) * **Xác định mục tiêu**. (Tốc độ xử lí - Đơn giản qui trình - Kết hợp các qui trình - Giảm lỗi trong đầu vào - Giảm lưu trữ dự phòng - Giảm dư thừa - Cải thiện hệ thống và tích hợp hệ thống con.) * **Lập kế hoạch và Kiểm soát Hoạt động**. (Kế hoạch bao gồm: Chon đội phân tích thiết kế hệ thống - Ước tính thời gian cần thiết để hoàn thành nhiệm vụ - lập kế hoạch dự án. - Kiểm soát dự án. So sánh các kế hoạch cho dự án với tiến hóa thực sự của nó. - Thực hiện hành động thích hợp để xúc tiến hoặc lên lịch các hoạt động.) * **Ước lượng thời gian – phân tích các điểm chức năng – yêu cầu nhân sự - quản lý rủi ro** * **Quản lí hoạt động phân tích và thết kế**. (Quản lý nhóm - Lắp ráp một đội - Chiến lược truyền thông nhóm - Mục tiêu năng suất của dự án - Động lực thành viên nhóm)   **SDLC – Agile (chương 4)**  **Prototyping** là một kỹ thuật thu thập thông tin hữu ích trong việc tìm kiếm các phản ứng, đề xuất, đổi mới của người dùng, và các kế hoạch sửa đổi.  **+ Patched-Up Prototype**: - Là một hệ thống hoạt động được nhưng theo kiểu chắp vá  -Có đầy đủ các tính năng nhưng không hiệu quả  -Người dùng có thể tương tác với hệ thống (làm quen với giao diện và loại đầu ra có sẵn)  - Thu thập và lưu trữ thông tin có thể không hiệu quả (các chức năng đã được viết nhanh chóng với mục tiêu là chạy được hơn là hiệu quả)  **+ Nonoperational Scale Models:**  - Một mô hình tỷ lệ không hoạt động mà được thiết lập để kiểm tra một số khía cạnh, chức năng của thiết kế  - Một bản mẫu với thông tin cho đầu vào và những yêu cầu đầu ra  Một mô hình tỷ lệ không hoạt động của một quy trình – hệ thống thông tin nhưng vì chi phí và thời gian quá mức cho phép nên sẽ không được tạo mẫu   * Người dùng có thể đưa ra quyết định dựa trên việc sử dụng nguyên mẫu đầu ra và đầu vào.   **+ First-of-a-Series Prototype:**  -Tạo ra một mô hình thử nghiệm  -Là một bản mẫu hoàn chỉnh  -Hữu dụng khi việc sửa đổi, cài đặt nhiều lần của hệ thống thông tin được lên kế hoạch  -Một nguyên mẫu hoàn chỉnh được cài đặt tại 1 hoặc 2 vị trí nếu thành công bản sao sẽ được cài đặt dựa trên mô hình sử dụng của khách hàng và một số yếu tố quan trọng  **+ Selected Features Prototype:**  - Tạo ra một mô hình hoạt động nhưng chỉ có một số tính năng mà hệ thống cuối cùng sẽ có (ví dụ một hệ thống có 6 chức năng nhưng chỉ có 3 chức năng hoạt động tốt)  -Được xây dựng thành các module (nếu các tính năng được được đánh giá thành công bởi người dùng, chúng có thể được kết hợp vào hệ thống thực tế ) | -Là một phần của hệ thống thực tế chứ không chỉ là một mô hình  **+Ưu điểm của SDLC**:  - Rút ngắn thời gian hiệu quả giữa việc xác định yêu cầu thông tin và việc cung cấp một hệ thống khả thi  -Vượt qua một số vấn đề để xác định chính xác yêu cầu thông tin người dùng.  **+Nhược điểm:**  -Định hình một hệ thống quá sớm trước cả khi vấn đề được hiểu rõ.  -Có thể dẫn đến việc tạo ra một hệ thống được chấp nhận bởi các nhóm người dùng cụ thể nhưng không đủ cho các nhu cầu của hệ thống tổng thể  **+Nhược điểm của Prototyping:**  -Khó khăn trong việc quản lý và phát triển đối với dự án lớn  -Người dùng và nhà phân tích có thể chấp nhận bản mẫu như là 1 hệ thống hoàn chỉnh nhưng thực tế là chưa đủ và không thể phục vụ như hệ thống hoàn chỉnh  **+Ưu điểm:**  -Thuận lợi trong việc thay đổi sớm trong quá trình phát triển hệ thống  -Cơ hội để dừng việc phát triển nếu nó không khả thi  -Phát triển hệ thống gần với yêu cầu và hi vọng của người dùng  **Các giai đoạn trong RAD:**  **+Lấy yêu cầu, lập kế hoạch:**  -Người dùng và nhà phân tích gặp nhau để xác định các yêu cầu, mục tiêu phát sinh của hệ thống  -Định hướng giải quyết các vấn đề kinh doanh (luôn duy trì mục tiêu kinh doanh)  **+RAD design workshop**:  -Thiết kế và lọc giai đoạn  -Sử dụng các phòng hệ thống hỗ trợ quyết định của nhóm nếu có  -Các nhà thiết kế sẽ lọc các module được thiết kế dựa trên phản hồi của người dùng  -Nếu những người sử dụng và các nhà phân tích có kinh nghiệm đều có mặt trong buổi workshop, không có nghi ngờ gì về nỗ lực sáng tạo này có thể thúc đẩy sự phát triển với tốc độ nhanh hơn.  **+Giai đoạn hiện thực:**  **So sánh RAD vs SDLC:**    **Ưu điểm của RAD:**  **Nhược điểm của RAD:**  -Cố gắng đẩy nhanh dự án quá nhiều, thiếu tài liệu, thiếu sự tham gia tốt của người dùng trong chu kỳ sống của phần mềm, Người phát triển phải có kỹ năng và được huấn luyện tốt cho việc dùng công cụ và thời gian phát triển nhanh, Hệ thống có  Khả năng phân tách modul, Cần có đâp ứng về thành phần sử dụng lại, Người phát triển và khách hàng phải nỗ lực, Người quản lý phải làm việc tận tụy với nhóm phát triển và khách hàng để nhanh chóng đạt được các thỏa thuận | **Agile Modeling: (plan, design, develop, test, evaluate, meet)**  **+ Nguyên tắc Agile: -** Cung cấp phản hồi nhanh - để cho con người hoặc hệ thống tạo ra mối liên hệ giữa kích thích và phản ứng, thông tin phản hồi phải xuất hiện trong khoảng thời gian hợp lý. -Giải quyết đơn giản - hơn 90% các vấn đề có thể được giải quyết với sự đơn giản hoàn toàn. -Thay đổi từng bước - bạn đang liên tục tạo ra sự thay đổi nhỏ nhất có thể mà vẫn tạo ra một sự khác biệt trong nỗ lực phát triển  -Nắm bắt thay đổi - chúng tôi muốn giữ tất cả các lựa chọn của chúng tôi mở, nhưng chúng tôi muốn để có thể đồng thời giải quyết bất cứ điều gì giới thiệu những trở ngại lớn nhất. -Khuyến khích công việc có chất lượng - tất cả những người tham gia đều muốn làm việc có chất lượng.  + **Tuyên ngôn của Agile:**  + **4 biến kiểm soát Agile**: thời gian, chi phí, chất lượng, phạm vi  + **4 thực hành Agile cốt lõi: -** Thời hạn phát hành ngắn: nhóm phát triển nén thời gian giữa các bản phát hành sản phẩm của họ.  -Tuần làm việc 40 giờ: nhóm phát triển cùng nhau thống nhất về văn hóa làm việc 40 giờ/ tuần  -Khách hàng tại chỗ: một người dùng chuyên gia trong lĩnh vực kinh doanh của công việc phát triển hệ thống sẽ luôn có mặt để trao đổi trong quá trình phát triển.  -Pair programming: 2 lập trình viên sẽ ngồi cùng nhau để làm việc  + **Quá trình phát triển Agile:** | **Phân tích độ phức tạp**  \*Nếu Q được thi công như một cấu trúc Heap, chúng ta có thể tạo ra một cấu trúc Heap trong bước khởi tạo (tức Q:= V[G]) với chi phí thời gian O(V) nếu xây dựng heap từ dưới lên. Vòng for khởi tạo mảng d và p tốn chi phí O(V).  Vòng while được thực hiện V lần và vì mỗi thao tác EXTRACT-MIN có độ phức tạp O(lgV), chi phí tính toán tất cả các lệnh gọi thao tác đó là O(VlgV)  Vòng for bên trong while được thực hiện O(E) lần vì tổng số nút của danh sách kế cận là E. Việc cập nhật thành phần d và p của đỉnh v trong heap tốn O(lgV). Vậy, tổng chi phí là O(V + V + VlgV + ElgV) = O(V+E)lgV  \*Nếu Q được thi công như một mảng, chúng ta có thể tạo ra một mảng Q trong bước khởi tạo (tức Q:= V[G]) với chi phí thời gian V.O(1)= O(V) . Vòng for khởi tạo mảng d và p tốn chi phí O(V).  Vòng while được thực hiện V lần và vì mỗi thao tác EXTRACT-MIN có độ phức tạp O(V), chi phí tính toán tất cả các lệnh gọi thao tác đó là O(V2)  Vòng for bên trong while được thực hiện O(E) lần vì tổng số nút của danh sách kế cận là E. Việc cập nhật thành phần d và p của đỉnh v trong mảng tốn O(1). Vậy, tổng chi phí là O(V + V2 + E) = O(V2+E)  BÀI TOÁN TÔ MÀU  **Độ phức tạp của giải thuật tô màu đồ thị**  + Giả sử đồ thị được biểu diễn bằng ma trận kế cận  + Trong thủ tục SAME\_COLOR mỗi ô trong ma trận kế cận đều được xem xét khi ta xét tô màu mới cho đỉnh chưa tô.  + Độ phức tạp của thủ tục SAME\_COLOR: O(n2) với n là số đỉnh của G.  + Nếu m là số màu được dùng để tô đồ thị thì thủ tục SAME\_COLOR được gọi tất cả m lần. Do đó, độ phức tạp của toàn giải thuật: m\* O(n2). Vì m thường là một số nhỏ, ta có thể nói:  ⇒ Giải thuật có độ phức tạp bậc 2  Ứng dụng: Xếp lịch thi học kì, gán tần số cho các đài vô tuyến. | **Chương 6:** **quay lui**  Bài toán con Mã và 8 con hậu  Cây không gian trạng thái  Một nút trên cây KGTT được gọi là *triển vọng* nếu nó tương ứng với lời giải bộ phận mà sẽ có thể dẫn đến lời giải đầy đủ; trái lại, nó được gọi là một lời giải *không triển vọng*.  Các nút lá diễn tả những trường hợp *bế tắc* (dead end) hay những *lời giải đầy đủ*.  Độ phức tạp: O(an)  Thời gian tính toán của các giải thuật quay lui thường là hàm mũ.  Nếu mỗi nút trên cây không gian trạng thái có trung bình *α* nút con, và chiều dài của lối đi lời giải là N, thì số nút trên cây sẽ tỉ lệ với αN.  Thời gian tính toán của giải thuật đệ quy tương ứng với số nút trên cây không gian trạng thái nên có độ phức tạp hàm mũ.  Giải thuật nhánh cận, áp dụng vào:  **Bài toán người thương gia du hành TSP** (chu trình Haminton với trọng số nhỏ nhất): khác với quay lui, áp dụng kĩ thuật này tùy thuộc vào việc tìm ra một hàm tính cận dưới thích hợp và thứ tự tạo ra các nút trên cây  **Bài toán cái túi dạng (0-1)**: có một đặc điểm khac biệt ở cây không gian trạng thái, mỗi nút trong cây biểu diễn một tập con của tập hợp mặt hang được cho.  ***Chương 8: Giải thuật xấp xỉ: đánh đổi tính tối ưu lấy tính dễ giải quyết - có độ phức tạp đa thức***  *Các cận về độ hữu hiệu của giải thuật xấp xỉ: max(c(i)/c\*(i), c\*(i)/c(i)) ≤ p(n) // (p(n) >=1, càng lớn càng tệ)*  *Đối với bài toán tối thiểu hóa chúng ta muốn tỷ số c(i)/c\*(i) càng min càng tốt.*  *Đối với bài toán cực đại hóa ngược lại.*  *Sai số tương đối |c(i) - c\*(i)|/ c\*(i)*  *Cận sai số tương đối: |c(i)-c\*(i)|/c\*(i)≤ ε(n)*  *Bài toán phủ đỉnh có cận tỷ số là 2*  *Bài toán phủ tập có cận tỷ số là ln|X|+1*  *Bài toán TSP: O(V2lgV), xấp xỉ cận tỷ số là 2* |
| **Đề xuất hệ thống (chương 7)**  - Là một sự tổng kết của tất cả những gì nhà phân tích hệ thống đã học được về người sử dụng, kinh doanh và về những gì cần thiết để cải thiện hiệu suất  - Là các phương pháp có hệ thống để:  + Có được phần cứng và phần mềm  + Xác định và dự báo chi phí và lợi ích  + Thực hiện phân tích lợi ích chi phí  Các bước được sử dụng để xác định nhu cầu phần cứng và phần mềm:  - Phần cứng máy tính lưu trữ hiện có sẵn  - Ước lượng công việc hiện tại và tương lai của hệ thống  - Đánh giá phần cứng và phần mềm có sẵn  - Chọn nhà cung cấp  - Mua thiết bị máy tính    - Kiểm kê Phần cứng máy tính: Loại thiết bị - số kiểu, nhà sản xuất, Trạng thái hoạt động của thiết bị - theo trình tự, trong kho, cần sửa chữa, Độ tuổi thiết bị ước tính, Tuổi thọ của thiết bị ,Vị trí vật lý của thiết bị, Phòng ban hoặc người chịu trách nhiệm về thiết bị, Tổ chức tài chính cho thiết bị - thuộc sở hữu, cho thuê, thuê.  - Ước lượng tải công việc: Các nhà phân tích hệ thống xây dựng số đại diện cho cả khối lượng công việc hiện tại và dự kiến cho hệ thống sao cho bất kỳ phần cứng nào thu được sẽ có khả năng xử lý khối lượng công việc hiện tại và tương lai. Hệ thống mới được đề xuất nên cắt giảm thời gian cần thiết cho con người và máy tính. --> Nếu ước tính được thực hiện đúng, doanh nghiệp không cần phải thay thế phần cứng chỉ do sự tăng trưởng không lường trước trong việc sử dụng hệ thống.  - Đánh giá phần cứng:  + Thời gian cần thiết cho các giao dịch trung bình - bao gồm thời gian cần để nhập dữ liệu và mất bao lâu để nhận được đầu ra.  + Tổng dung lượng thể tích của hệ thống - có thể xử lý bao nhiêu cùng một lúc trước khi có vấn đề phát sinh.  + Thời gian nhàn rỗi của CPU hoặc mạng  + Kích thước bộ nhớ được cung cấp  - Những người đánh giá phần cứng: Người quản lý, Người sử dụng, Nhà phân tích hệ thống. Đánh giá phần cứng máy tính là trách nhiệm chung của quản lý, người dùng và các nhà phân tích hệ thống. Mặc dù các nhà cung cấp cung cấp chi tiết về các dịch vụ của họ, các nhà phân tích giám sát việc đánh giá cá nhân. Các nhà phân tích của hệ thống sẽ giáo dục người dùng và quản trị về các lợi thế và bất lợi.  - Mua lại thiết bị máy tính: Mua, cho thuê, thuê.  + Các yếu tố ảnh hưởng: Chi phí ban đầu so với chi phí dài hạn, Có thể huy động vốn để gắn kết trong thiết bị máy tính, Nếu doanh nghiệp có toàn quyền kiểm soát và chịu trách nhiệm về thiết bị máy tính.  **Mua**: Khi các hệ thống trở nên nhỏ hơn, mạnh hơn, ít tốn kém, và các hệ thống phân phối ngày càng trở nên phổ biến, nhiều doanh nghiệp đang quyết định mua thiết bị.  - Lợi: Rẻ hơn cho thuê hoặc cho vay trong thời gian dài, Khả năng thay đổi hệ thống, Cung cấp lợi thế thuế của khấu hao nhanh, Toàn quyền điều khiển  **Thuê:** Do chi phí cao liên quan và thực tế là công ty sẽ không sở hữu thiết bị thuê, thuê nên được xem như là một di chuyển ngắn hạn để xử lý các nhu cầu máy tính không chính xác hoặc hạn chế hoặc thời gian công nghệ không ổn định.  **+ Tạo phần mềm tùy chỉnh:** Lợi: Phản ứng cụ thể đối với các nhu cầu kinh doanh chuyên biệt, Đổi mới có thể tạo ra lợi thế cạnh tranh, Nhân viên trong nhà có sẵn để duy trì phần mềm, Tự hào về quyền sở hữu / Hại: Có thể có chi phí ban đầu cao hơn đáng kể so với phần mềm COTS hoặc ASP, Sự cần thiết của việc tuyển dụng hoặc làm việc với một nhóm phát triển, phải duy trì.  **- Mua gói COTS (Phần mềm thương mại):** Xem xét việc sử dụng phần mềm COTS khi bạn có thể dễ dàng tích hợp các ứng dụng hoặc gói vào các hệ thống hiện có hoặc đã được lên kế hoạch và khi bạn xác định không cần thiết phải thay đổi hoặc tùy biến chúng cho người dùng một cách ngay lập tức hoặc liên tục. Ít hơn một nửa của tất cả các dự án được phát triển từ các ứng dụng hiện có hoặc các thành phần. Ít hơn 5 phần trăm của phần mềm là phần mềm miễn phí mà không yêu cầu sửa đổi nào cả. | **Lợi**: Tinh tế trong thế giới thương mại, Tăng độ tin cậy, Tăng chức năng, Thường chi phí ban đầu thấp hơn, Đã được sử dụng bởi các công ty khác, Trợ giúp và đào tạo đi kèm với phần mềm / **Hại**: Tập trung lập trình ; không tập trung kinh doanh, Phải sống với các tính năng hiện có, Tùy chỉnh hạn chế, Tương lai tài chính không chắc chắn của nhà cung cấp, Ít quyền sở hữu và cam kết  **- Cung cấp bởi nhà cung cấp dịch vụ:** Lợi: Các tổ chức không chuyên về các hệ thống thông tin có thể tập trung vào những gì họ làm tốt nhất (sứ mệnh chiến lược của họ), Không cần thuê, đào tạo, hoặc giữ lại một nhân viên CNTT lớn, Không có chi tiêu của nhân viên thời gian vào các nhiệm | vụ CNTT không cần thiết / Hại: Mất kiểm soát dữ liệu, hệ thống, nhân viên IT, và lịch biểu, Mất kiểm soát dữ liệu, hệ thống, nhân viên IT, và lịch biểu, Mối quan tâm đến tính khả thi về tài chính và sự ổn định lâu dài của ASP, An ninh, bảo mật, và mối quan tâm riêng tư, Mất lợi thế chiến lược tiềm năng của công ty liên quan đến tính đổi mới của ứng dụng  **Công cụ Hỗ trợ Quyết định:** AHP và các phần mềm đa tiêu chuẩn khác, Hệ thống chuyên gia và mạng lưới thần kinh, Hệ thống khuyến nghị, Lấy thông tin bên ngoài từ Web  **Ước lượng Xu hướng**: Thẩm định đồ họa, Phương pháp của hình vuông nhỏ nhất, Trung bình động  - Thẩm định đồ họa: Nhìn vào đồ thị và ước lượng bằng tay một phần mở rộng của một đường hoặc đường cong.Sự đánh giá đồ họa là cách đơn giản nhất để xác định xu hướng và dự báo xu hướng trong tương lai.  - Phương pháp của hình vuông nhỏ nhất: Tìm đường phù hợp nhất bằng cách giảm thiểu tổng các sai lệch từ đường, Một khi đã tìm được đường dây phù hợp nhất, có thể mở rộng để dự báo điều gì sẽ xảy ra  - Trung bình động: Tính trung bình số học của dữ liệu từ các nhóm hoặc các khoảng thời gian, sau đó tính toán số trung bình kế tiếp bằng cách loại bỏ dữ liệu của khoảng thời gian lâu nhất và thêm vào giai đoạn tiếp theo.  **Xác định lợi ích và chi phí**: Hữu hình, vô hình.  - Lợi ích hữu hình: Ưu điểm có thể đo được bằng đô la thu được cho tổ chức thông qua việc sử dụng hệ thống thông tin. Vd: Tăng tốc độ xử lý, Truy cập thông tin không thể truy cập được, Tiếp cận thông tin trên cơ sở kịp thời hơn  - Lợi ích vô hình: là lợi ích từ việc sử dụng hệ thống thông tin khó đo lường. Vd: Cải thiện quá trình ra quyết định, Nâng cao độ chính xác, Trở nên cạnh tranh hơn trong dịch vụ khách hàng  - Chi phí hữu hình:Những người có thể được dự đoán chính xác bởi các nhà phân tích hệ thống và nhân viên kế toán của doanh nghiệp.Vd:Chi phí thiết bị,Chi phí tài nguyên,chi phí của các nhà phân tích thời gian của hệ thống  - Chi phí vô hình: Những thứ rất khó ước tính, và có thể không được biết đến. Vd: Mất cạnh tranh, Mất danh tiếng vì là người đầu tiên, Giảm hình ảnh công ty  - So sánh chi phí và lợi ích: Phân tích chiết khấu, Hoàn vốn, Phân tích dòng tiền mặt, Phân tích giá trị hiện tại  **- Phân tích chiết khấu:** Điểm tại đó tổng chi phí của hệ thống hiện tại và hệ thống đề xuất giao nhau, Có ích khi một doanh nghiệp đang phát triển và khối lượng là một yếu tố chính trong chi phí. Tổng chi phí = chi phí phát sinh trong quá trình hoạt động của hệ thống + chi phí phát triển chỉ xảy ra một lần. Lợi: Có thể xác định mất bao lâu để các lợi ích của hệ thống trả lại chi phí phát triển nó / Hại: Lợi ích được giả định là vẫn như cũ.  **- Phân tích dòng tiền mặt**: Xem xét hướng, quy mô và mô hình dòng tiền mặt gắn với hệ thống thông tin được đề xuất, được sử dụng để xác định thời điểm một công ty sẽ bắt đầu kiếm được lợi nhuận và khi nào nó sẽ không có lợi nhuận.  **- Phân tích giá trị hiện tại:** Cách để đánh giá tất cả các khoản chi tiêu kinh tế và thu nhập của hệ thống thông tin trong đời sống kinh tế của nó và so sánh chi phí ngày nay với chi phí trong tương lai và lợi ích hiện tại với lợi ích trong tương lai  **- Hướng dẫn Phân tích:** Sử dụng phân tích chiết khấu nếu dự án cần phải được chứng minh về chi phí chứ không phải lợi ích.Sử dụng hoàn vốn khi lợi ích hữu hình được cải thiện tạo thành một lập luận thuyết phục cho hệ thống đề xuất. Sử dụng phân tích dòng tiền mặt khi dự án tốn kém, so với quy mô của công ty. Sử dụng giá trị hiện tại khi thời gian hoàn vốn dài hoặc khi chi phí vay vốn cao | - Các loại đồ thị: đường, cột, thanh, hình tròn.  - Biểu đồ đường: Được sử dụng để thể hiện sự thay đổi theo thời gian, Thay đổi tối đa năm biến trên một biểu đồ đơn, Cũng có thể hiển thị khi đường giao nhau    **-** Biểu đồ cột: Có thể mô tả sự so sánh giữa hai hay nhiều biến theo thời gian, Được sử dụng thường xuyên hơn để so sánh các biến số khác nhau tại một điểm cụ thể trong thời gian, Dễ hiểu hơn đồ thị dòng | **Chương 6:** **quay lui**  Bài toán con Mã và 8 con hậu  Cây không gian trạng thái  Một nút trên cây KGTT được gọi là *triển vọng* nếu nó tương ứng với lời giải bộ phận mà sẽ có thể dẫn đến lời giải đầy đủ; trái lại, nó được gọi là một lời giải *không triển vọng*.  Các nút lá diễn tả những trường hợp *bế tắc* (dead end) hay những *lời giải đầy đủ*.  Độ phức tạp: O(an)  Thời gian tính toán của các giải thuật quay lui thường là hàm mũ.  Nếu mỗi nút trên cây không gian trạng thái có trung bình *α* nút con, và chiều dài của lối đi lời giải là N, thì số nút trên cây sẽ tỉ lệ với αN.  Thời gian tính toán của giải thuật đệ quy tương ứng với số nút trên cây không gian trạng thái nên có độ phức tạp hàm mũ.  Giải thuật nhánh cận, áp dụng vào:  **Bài toán người thương gia du hành TSP** (chu trình Haminton với trọng số nhỏ nhất): khác với quay lui, áp dụng kĩ thuật này tùy thuộc vào việc tìm ra một hàm tính cận dưới thích hợp và thứ tự tạo ra các nút trên cây  **Bài toán cái túi dạng (0-1)**: có một đặc điểm khac biệt ở cây không gian trạng thái, mỗi nút trong cây biểu diễn một tập con của tập hợp mặt hang được cho.  ***Chương 8: Giải thuật xấp xỉ: đánh đổi tính tối ưu lấy tính dễ giải quyết - có độ phức tạp đa thức***  *Các cận về độ hữu hiệu của giải thuật xấp xỉ: max(c(i)/c\*(i), c\*(i)/c(i)) ≤ p(n) // (p(n) >=1, càng lớn càng tệ)*  *Đối với bài toán tối thiểu hóa chúng ta muốn tỷ số c(i)/c\*(i) càng min càng tốt.*  *Đối với bài toán cực đại hóa ngược lại.*  *Sai số tương đối |c(i) - c\*(i)|/ c\*(i)*  *Cận sai số tương đối: |c(i)-c\*(i)|/c\*(i)≤ ε(n)*  *Bài toán phủ đỉnh có cận tỷ số là 2*  *Bài toán phủ tập có cận tỷ số là ln|X|+1*  *Bài toán TSP: O(V2lgV), xấp xỉ cận tỷ số là 2* |