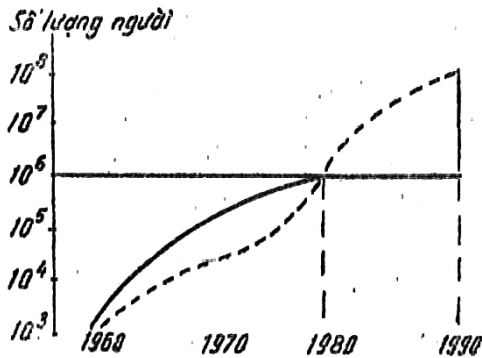


## CÔNG NGHỆ LẬP TRÌNH VÀ TRÍ TUỆ NHÂN TẠO

*Lịch sử của bất kỳ một công nghệ nào cũng kết xuất từ hỗn hợp các vọng tưởng, các lý thuyết và thực tiễn» (G.L.SIMON)*

Mặc dầu Charles Ballage và Ada Lovelace đã viết những dòng chương trình đầu tiên cho máy tính từ giữa thế kỷ 19, nhưng việc lập trình chỉ được coi như là một nghề chuyên môn trong khoảng 40 năm trở lại đây. Theo thống kê, số lượng thảo chương viên chuyên nghiệp ở Tây Âu và Mỹ đã lên tới con số hàng triệu với tổng giá trị sản phẩm chương trình hàng tỉ đô la vào giữa thập niên 80. Tuy vậy, một điều đáng ngạc nhiên là công việc lập trình thời gian qua chủ yếu vẫn mang nặng tính thủ công, đơn chiếc, trong khi công nghệ chế tạo máy tính có những bước tiến nhảy vọt tạo ra các thế hệ máy ngày càng hoàn thiện - đánh dấu từng giai đoạn phát triển của Tin học. Thế giới ngày nay đang đứng trước ngưỡng cửa của một cuộc cách mạng tin học, một nền văn hóa tin học mà mỗi con người sẽ đều chung hưởng thành quả và tham gia tác động một cách tự nhiên vào các quá trình cải biến kinh tế - xã hội. Xây dựng và hoàn thiện công nghệ lập trình cho các máy tính thế hệ mới có sử dụng những thành tựu của Trí tuệ nhân tạo là một trong những yếu tố quan trọng đảm bảo thành công của sự nghiệp Tin học hóa con người - xã hội vào cuối thế kỷ này.



— : đường biểu diễn số lượng thảo chương viên chuyên nghiệp theo từng thập niên  
 - - - : đường biểu diễn số lượng người sử dụng máy tính theo từng thập niên (thảo chương viên không chuyên nghiệp)

Biểu đồ phản ánh mức độ Tin học hóa ở Mỹ theo khía cạnh số lượng người tham gia lập trình và sử dụng máy tính (1960-1990)

### I-CÔNG NGHỆ LẬP TRÌNH CHO TRÍ TUỆ NHÂN TẠO

Nếu như lịch sử phát triển của Tin học thường được ghi nhận qua các tiến bộ kỹ thuật

của công nghệ chế tạo máy tính thì có thể tương ứng quá trình tin học hóa đã diễn biến tích cực theo từng cấp độ quan hệ giữa người và máy tính như sau :

- Cấp độ 1 (thập niên 50) : Điều hành viên - máy tính
- Cấp độ 2 (thập niên 60) : Thảo chương viên - máy tính
- Cấp độ 3 (thập niên 70) : Phân tích viên - máy tính
- Cấp độ 4 (thập niên 80) : Người sử dụng làm việc trực tiếp với máy tính

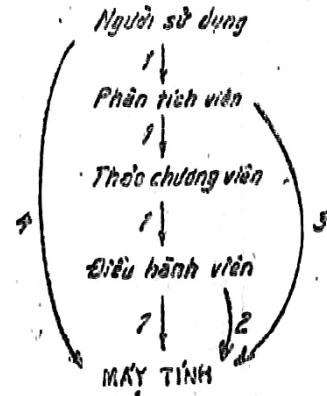
Ở những cấp độ đầu, công việc lập trình hướng về máy - đòi hỏi thời gian đào tạo và rèn luyện kỹ năng khá công phu cho việc sử dụng máy thông qua một đội ngũ chuyên viên từ điều hành tới phân tích - đã hạn chế không ít đến năng suất, hiệu quả và khả năng sử dụng rộng rãi các máy tính. Xu thế hướng về người sử dụng, cùng với những thành tựu trong quá trình nâng cao trí tuệ máy đã đổi mới và làm phong phú một loạt khái niệm cơ bản của Tin học như chương trình, thuật giải, dữ liệu, v.v... đồng thời cũng dẫn đến những nhu cầu tự nhiên của việc xây dựng, hoàn thiện công nghệ lập trình nói chung và công nghệ lập trình định hướng vào Trí tuệ nhân tạo với những nội dung chủ yếu sau đây :

- Công nghệ lập trình cho các hệ xử lý và giải thích tín hiệu (ảnh, tiếng nói, các tín hiệu ở dải sóng đặc biệt, v.v...).

- Công nghệ lập trình cho các hệ xử lý và khai thác cơ sở tri thức.
- Công nghệ lập trình cho các hệ học và robot thông minh.

Sơ đồ biểu diễn quan hệ người-máy qua các thập niên

- 1 : thập niên 50
- 2 : thập niên 60
- 3 : thập niên 70
- 4 : thập niên 80



**1. Công nghệ lập trình cho các hệ xử lý và giải thích tín hiệu**

Các sản phẩm chương trình trong lĩnh vực này định hướng vào các vấn đề chủ yếu sau đây :

1. Xử lý và giải thích thông tin đồ họa :
  - Hiện thị thông tin đồ họa 2,3 và nhiều chiều,
  - Nhận dạng ký hiệu,
  - Biến đổi và tạo hình-(trò chơi, quảng cáo...)

**2. Xử lý và giải thích ảnh :**

- Xử lý, giải đoán ảnh viễn thám, điều tra cơ bản, nghiên cứu cơ bản,
- Xử lý nhận dạng ảnh công nghiệp, dân dụng
- Xử lý, nhận dạng văn bản, vân tay,...

**3. Phân tích và tổng hợp hình ảnh :**

- Tự động hóa thiết kế,
- Phân tích và tổng hợp bản đồ,
- Mô phỏng thực nghiệm,
- Phim hoạt họa.

**4. Xử lý và nhận dạng tiếng nói.**

**5. Phân tích và tổng hợp tiếng nói.**

Các ngữ trình thông dụng bao gồm : BASIC, PASCAL, LOGO, FORTRAN, ASSEMBER, GIPSY, C, BALSAL, ...

Mức độ công nghệ và sản phẩm : hầu như chưa có phần mềm công cụ ; các sản phẩm cuối cùng ở dạng paket (1) và các hệ trình gắn với loại máy xác định : việc chuyên đổi, cải biên mở rộng, hoàn thiện sản phẩm đòi hỏi nhiều công phu và kỹ năng chuyên biệt. Tuy vậy, cũng đã có một số nghiên cứu nhằm xác định và hoàn thiện công nghệ dựa trên việc xây dựng các ngữ trình chuẩn, chuẩn hóa các thiết kế hệ thống, phần mềm công cụ tương ứng cho từng vấn đề. Ngữ trình chuẩn dự kiến : cần đảm bảo khả năng xử lý phối hợp linh hoạt các thiết bị vào, ra tín hiệu ; tính toán và xử lý song song ; tương hợp và dung nạp được các hệ thống xử lý đang phổ dụng. Một số bản dạng đưa ra theo kiểu ngữ trình MANDALA, OCCAM, COMPEL.

Phần mềm công cụ và chuẩn hóa thiết kế hệ thống đã được tiến hành xây dựng cho các vấn đề A1, A2, A3 : Graphic Kernel System (GKS), Core Graphic System (CGS), Standard Image Processing System (SIPS) và một số hệ cho CAD, CAM, CAE, CAP, CAT, COM.

CAD : computer aided design,

CAM : » manufacturing,

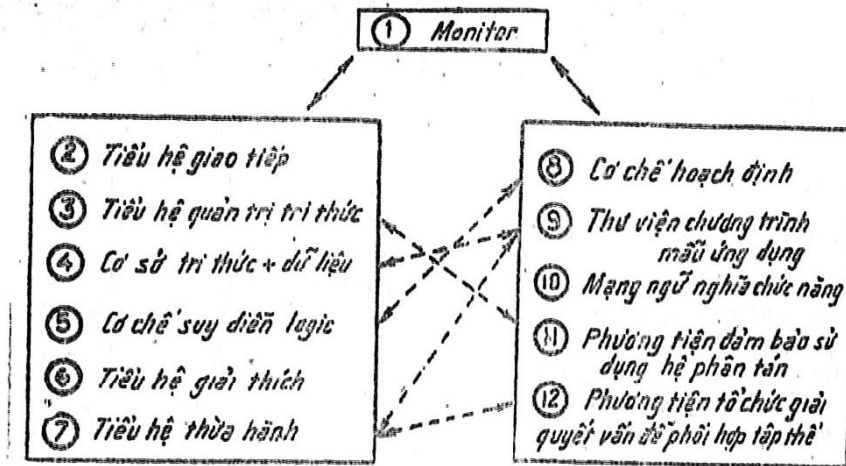
CAP : » planning,

CAT : » testing,

COM : computer output on Microfilm or Microfiche.

(1) Các bộ chương trình

## 2. Công nghệ lập trình cho các hệ xử lí và khai thác trí thức



Sơ đồ tổng quát cho các hệ xử lí và khai thác trí thức

Theo sơ đồ này, các hệ liên quan đến cơ sở tri thức sẽ tương ứng với từng tổ hợp (toàn bộ hoặc rút gọn) các thành phần từ 1 đến 12 như sau:

- |                                       |   |                 |
|---------------------------------------|---|-----------------|
| 1. Hệ tìm kiếm (thông tin) thông minh | : | ① ② ③ ④ ⑤       |
| 2. Hệ giải toán thông minh            | : | ① ② ④ ⑧ ⑨ ⑩     |
| 3. Hệ chuyên gia                      | : | ① ② ③ ④ ⑤ ⑥ ⑦   |
| 4. Hệ chuyên gia phân tán             | : | ① ② ③ ④ ⑤ ⑥ ⑦ ⑪ |
| 5. Hệ chuyên gia sử dụng tập thể      | : | ① ② ③ ④ ⑤ ⑥ ⑦ ⑫ |
| 6. Hệ chuyên gia tổng quát            | : | ① → ⑫           |

Các ngữ trình thông dụng bao gồm: FORTRAN, PASCAL, LISP, PROLOG, C, PL/I, ADA.

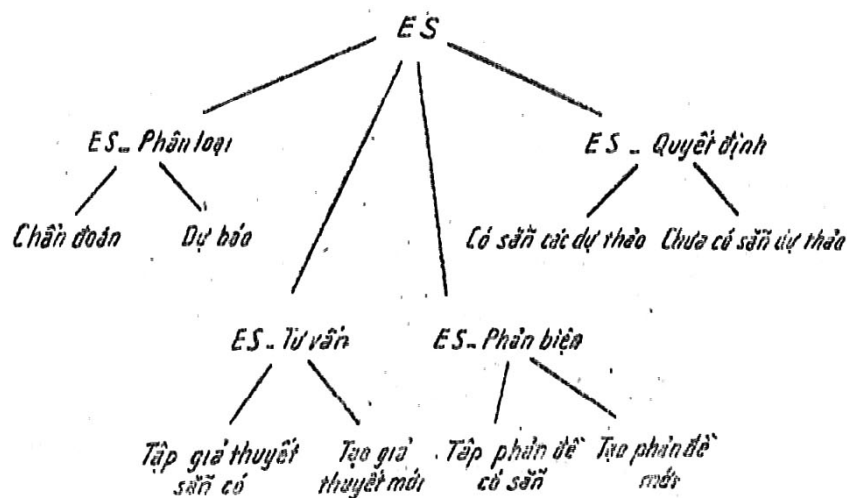
Mức độ công nghệ và sản phẩm: Đã hình thành phần mềm công cụ xây dựng cơ sở tri thức và hệ chuyên gia theo các phương án hoàn thiện Hệ tự động hóa thiết kế các hệ chuyên gia; hệ chuyên gia độc lập với vấn đề (Hệ chuyên gia với cơ sở tri thức rỗng); các hệ phân tích, tiếp nhận và chuyển đổi biểu diễn tri thức. Các sản phẩm chủ yếu vẫn là các hệ sử dụng cơ sở tri thức định hướng theo vấn đề và chưa thuận tiện cho người sử dụng. Ngữ trình chuẩn dự kiến: Cần đảm bảo khả năng phối hợp giữa thao tác lập luận và tính toán, thuận tiện cho việc biểu diễn, xử lý tri thức mô tả cũng như tri thức thủ tục, ... Một số bản dạng ban đầu đã định hướng vào việc kết hợp LISP và PROLOG (LISPLOG) hoặc ngôn ngữ tựa C.

Xu hướng sắp tới tập trung vào hoàn thiện các phần mềm công cụ tiếp nhận, sàng lọc tri thức; các hệ đảm bảo cơ chế lập luận chính xác và xấp xỉ; các tiêu hệ đảm bảo giao tiếp thuận tiện giữa người sử dụng (không phải chuyên gia) với hệ chuyên gia trên cơ sở ngôn ngữ tự nhiên.

## 3. Công nghệ lập trình cho các hệ học và Robot thông minh

Sản phẩm chương trình đầu tiên cho Robot đã được ê-kíp Stanford (do N.Nilsson và B.Raphael lãnh đạo) hoàn thành vào năm 1967 cùng với bản dạng đầu tiên về hệ học của Rosenblatt trước đó vài năm đã mở ra một lĩnh vực lập trình mới bao trùm lên những vấn đề lằng mạn nhất, gây nhiều tranh luận nhất suốt hai thập niên qua. Tuy nhiên, khối lượng sản phẩm chương trình cho các hệ học và Robot cũng mới chỉ tập trung vào một số kiểu loại sau đây:

- Hệ « học vẹt » và Robot điều khiển theo chương trình cứng nhắc (thế hệ 1).
- Hệ « học theo lời khuyên » và Robot điều khiển theo chương trình linh hoạt (thế hệ 2).



Sơ đồ phân loại kiểu hệ chuyên gia theo nội dung công việc

- Hệ « học theo ví dụ » và Robot thông minh (thế hệ 3).
- Các phần mềm bảo đảm cho việc tiếp nhận và hiểu tri thức qua hình ảnh và ngôn ngữ tự nhiên (viết, nói).

Các ngữ trình đã được dùng bao gồm: PASCAL, VAL, CAR, COLA.

Phần mềm công cụ cũng đã hình thành một số bản dạng ban đầu:

- ARC (Advanced Robot Control Software).
- PLANNER, GENERA, LAOCON, PROBESTA, META-DENDRAL...

Hiện nay ngữ trình chuẩn dự kiến cho các hệ này cũng chưa xác định do tính phức tạp của việc kết hợp đồng thời các khả năng xử lý tri thức ngôn ngữ, thông tin hình ảnh, phối hợp với các lệnh thao tác, điều khiển song hành v.v. .

Xu hướng sắp tới sẽ tiếp tục hoàn thiện công nghệ lập trình cho các Hệ học và Robot cả 3 thế hệ nêu trên, đồng thời định hình rõ các thiết kế chuẩn và hệ mềm công cụ cho Hệ học và Robot thế hệ 4 trên cơ sở các đảm bảo lý thuyết về Hệ học, các phương tiện kỹ thuật tiên tiến nhất. Chắc chắn là thập kỷ 90 sẽ được chứng kiến sự xuất hiện hàng loạt các Hệ học và Robot thế hệ mới mà sự ra đời không những có ảnh hưởng to lớn đến tiến bộ khoa học kỹ thuật và xã hội nói chung mà còn tạo ra những đổi mới sâu sắc trong công nghệ lập trình dẫn đến sự hình thành Công nghệ lập trình theo những nguyên lý của trí tuệ nhân tạo.

## II--MỘT SỐ ĐÓNG GÓP CỦA TRÍ TUỆ NHÂN TẠO CHO CÔNG NGHỆ LẬP TRÌNH

Như nhiều chuyên gia tin học đã nhận định: « Cho đến cuối thập niên 80 của thế kỷ này, công nghệ lập trình chưa có những cải cách về cơ bản, Những gì mà chúng ta đang làm hôm nay là để chuẩn bị cho bước nhảy vọt của công nghệ lập trình—trong đó Trí tuệ nhân tạo đóng vai trò rất quan trọng—vào đầu thập niên 90...» (Báo cáo « Dự án máy tính thế hệ 5 », Nhóm Carnegie Melon).

Ngoài những thành tựu của trí tuệ nhân tạo đóng góp chung cho việc cải thiện, đổi mới môi trường tin học đã dẫn đến những hiệu ứng phụ rất tự nhiên cho việc biến cải tích cực các phương tiện giao tiếp thuận lợi đối với công việc lập trình: Tự động hóa việc đọc văn bản trực tiếp, soạn thảo văn bản trực tiếp trên cơ sở các thiết bị, biểu thị hình ảnh và nhận dạng, hiểu ngôn ngữ tự nhiên (viết, nói)... Chính những hiệu ứng phụ này lại tác động vào việc hoàn thiện, phong phú thêm các phương pháp lập trình; đòi hỏi các sản phẩm chương trình những yêu cầu mới, những chức năng mới định hình dần cho công nghệ lập trình mới.

Những đóng góp trực tiếp nhất của Trí tuệ nhân tạo cho công nghệ lập trình bao gồm các nội dung sau:

1, Hoàn thiện và đổi mới những quan niệm về chương trình, phương tiện lập trình, phong cách lập trình trên cơ sở các thành tựu của trí tuệ nhân tạo trong việc giải quyết các vấn đề liên quan đến nâng cao trí tuệ máy.

Alan Kay đã nhấn mạnh vai trò của 4 lần cải biến LISP trong quãng thời gian 1960--1980 đã dẫn đến những đổi thay về ngữ trình và phong cách lập trình:

- Các ngữ trình thủ tục mở rộng thêm các ngữ trình không thủ tục (không tách biệt giữa dữ liệu và lệnh thao tác trên dữ liệu như các ngữ trình truyền thống).

- Phong cách lập trình chuyển từ phong cách thủ tục (How) sang phong cách mô tả (What) (không sa vào chi tiết tỉ mỉ làm như thế nào mà chỉ cần xác định rõ những điều gì cần phải thực hiện để giải một vấn đề cho trước).

- Phương tiện lập trình gián tiếp đã được bổ xung các phương tiện trực tiếp, dễ dàng thuận lợi cho việc soạn thảo, kiểm tra, hoàn tất chương trình.

2, Tự động hóa công việc lập trình trên cơ sở các hệ học và các hệ sử dụng cơ sở tri thức lập trình. Hiện tại đang hình thành các hướng phát triển:

- Các hệ xây dựng, tổng hợp chương trình từ ví dụ, đặc tả, kịch bản.

- Các hệ biến đổi, rút gọn tối ưu hóa chương trình trên cơ sở các hệ học theo qui nạp và tương tự.

- Các hệ tự động hóa dịch chương trình theo nguyên lý hệ chuyên gia.

Cho đến nay đã tồn tại các hệ dịch tự động từ các ngôn ngữ bậc cao giữa 2 hệ máy (FORTRAN--FORTRAN; PL/1--PL/1; FORTRAN, PL/1, LIST--ADA, C; v.v...).

Shamos, giám đốc hãng Lexème đã thông báo vấn đề dịch chương trình từ ngữ trình cấp thấp sang ngữ trình cấp cao sẽ được giải quyết về căn bản vào cuối thập niên 80 này. Các thẩm định sơ bộ cho biết chi phí dịch máy chỉ bằng 20% chi phí dịch bởi chuyên gia lập trình.

Trên nhiều tài liệu nghiêm túc đã nhắc đến thuật ngữ «Robot lập trình»--một khái niệm tổng hòa phân ánh rất sinh động mới liên hệ tự nhiên và tất yếu giữa công nghệ lập trình và trí tuệ nhân tạo, đã bắt đầu hình thành gần như cùng quãng thời gian ra đời dự án máy tính thế hệ 5. Tin học ngày mai đang được chuẩn bị thế đứng mới trên cái kiềng 3 chân; máy tính thế hệ mới, công nghệ lập trình mới và trí tuệ nhân tạo. Chúng ta hy vọng rằng, như Alan Kay và G.Simon đã tiên đoán, phản ứng dây chuyền sẽ bùng nổ: sau cuộc cách mạng về công nghệ lập trình sẽ dẫn đến cuộc Cách mạng tin học mà Trí tuệ nhân tạo luôn luôn là tác nhân chính trong cả 2 cuộc cách mạng này--chắc chắn tạo ra những biến chuyển sâu sắc về kinh tế--xã hội trên phạm vi toàn thế giới.

HOÀNG KIỂM

Hội thảo Công nghệ lập trình

Bãi Cháy - 8/87

Nhận ngày 25-9-1987