

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**



ISO 9001:2008

**TRẦN NHẬT GIÁP**

**LUẬN VĂN THẠC SĨ**  
**NGÀNH HỆ THỐNG THÔNG TIN**

**Hải Phòng - 2016**

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

TRẦN NHẬT GIÁP

***ĐÁNH GIÁ DỰ ÁN ĐẦU TƯ VÀ  
LẬP LỊCH QUẢN LÝ DỰ ÁN TỰ ĐỘNG***

LUẬN VĂN THẠC SĨ  
NGÀNH CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN  
MÃ SỐ: 60 48 01 04

NGƯỜI HƯỚNG DẪN KHOA HỌC:  
PGS.TS. NGUYỄN VĂN VỸ

## MỤC LỤC

MỤC LỤC.....	1
BẢNG CÁC CHỮ VIẾT TẮT.....	5
BẢNG DANH MỤC CÁC HÌNH.....	6
BẢNG DANH MỤC CÁC BẢNG .....	8
LỜI CẢM ƠN .....	8
LỜI CAM ĐOAN .....	9
MỞ ĐẦU.....	10
1. Đặt vấn đề.....	10
2. Đối tượng và phạm vi nghiên cứu.....	11
2.1. Đối tượng nghiên cứu.....	11
2.2. Phạm vi nghiên cứu.....	11
3. Hướng nghiên cứu của đề tài .....	12
4. Những nội dung nghiên cứu chính.....	12
5. Phương pháp nghiên cứu.....	12
6. Ý nghĩa khoa học của đề tài .....	12
CHƯƠNG I: THỰC TRẠNG VỀ QUẢN LÝ DỰ ÁN VÀ LẬP LỊCH TRONG QUẢN LÝ DỰ ÁN .....	13
1.1. Khái quát về quản lý dự án.....	13
1.1.1. Định nghĩa dự án .....	13
1.1.2. Đánh giá khái quát để lựa chọn dự án.....	14
1.1.3. Đánh giá khả thi kinh tế của dự án.....	14
1.2. Lập kế hoạch dự án và bài toán lập lịch.....	15
1.2.1. Sơ đồ tổng thể lập kế hoạch dự án .....	15
1.2.2. Các khó khăn của việc lập kế hoạch dự án và bài toán lập lịch.....	18
1.2.3. Một số phần mềm đã sử dụng để lập lịch.....	18
CHƯƠNG II: ĐÁNH GIÁ KHẢ THI VÀ LẬP KẾ HOẠCH LỊCH THỜI GIAN BẰNG TAY .....	20
2.1. Đánh giá khả thi kinh tế của dự án.....	20
2.1.1. Sơ đồ thực hiện đánh giá khả thi kinh tế của dự án .....	20
2.1.2. Tính toán hệ số hoàn vốn và thời gian hoàn vốn .....	21
2.2. Lập kế hoạch lịch thời gian cho dự án .....	23
2.2.1. Thuật toán lập mạng AOA bằng tay .....	24
2.2.2. Sơ đồ khái niệm của thuật toán lập mạng bằng tay.....	26
2.2.3. Ví dụ minh họa thuật toán lập mạng bằng tay .....	29

2.2.4. Sử dụng mạng lập được để lập lịch dự án .....	34
<b>Chương III: THIẾT KẾ THUẬT TOÁN CHO VIỆC TỰ ĐỘNG TÍNH TOÁN DỰ</b>	
<b>ÁN .....</b>	<b>37</b>
3.1. Tính toán đánh giá khả thi kinh tế của dự án .....	37
3.1.1. Các tham số để tính toán hệ số hoàn vốn .....	37
3.1.2. Cấu trúc bảng tính toán phân tích khả thi kinh tế .....	37
3.1.3. Ví dụ tính toán phân tích khả thi kinh tế .....	39
3.2. Chuyển thuật toán lập kế hoạch dự án làm tay sang làm máy .....	40
3.2.1. Sơ đồ tổng quát chuyển đổi thuật toán sang làm máy .....	40
3.2.2. Bảng cấu trúc dữ liệu cho thuật toán lập kế hoạch dự án trên máy .....	40
3.2.3. Thiết kế thuật toán cho chương trình lập mạng AOA .....	42
3.2.4. Sơ đồ logic tính các tham số thời gian trên mạng AOA .....	51
3.2.5. Sơ đồ logic vẽ các biểu đồ của kế hoạch lịch .....	53
3.2.6. Giới thiệu về chương trình lập mạng cho kế hoạch lịch .....	54
3.2.7. Một số ví dụ thử nghiệm sử dụng chương trình thuật toán .....	55
<b>Chương IV: XÂY DỰNG CHƯƠNG TRÌNH TRỢ GIÚP QUẢN LÝ DỰ ÁN .....</b>	<b>60</b>
4.1. Bài toán quản lý, điều hành dự án và chương trình trợ giúp .....	60
4.2. Thiết kế dữ liệu vật lý cho chương trình .....	61
4.3. Giới thiệu chương trình trợ giúp quản lý dự án .....	64
4.3.1. Hệ thống thực đơn .....	64
4.3.2. Một số chức năng chính của chương trình và giao diện .....	65
4.3.3. Một ví dụ thực hiện dự án cụ thể với chương trình .....	68
<b>KẾT LUẬN .....</b>	<b>76</b>
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>78</b>
<b>PHỤ LỤC .....</b>	<b>80</b>
<b>A. PHỤ LỤC 1: Một số kết quả tính toán của chương trình lập lịch .....</b>	<b>80</b>
A1.1. Ví dụ 1 .....	80
A1.2. Ví dụ 2 .....	83
A1.3. Ví dụ 3 .....	86
A1.4. Ví dụ 4 .....	89
<b>B. PHỤ LỤC 2. Mã nguồn chương trình trợ giúp quản lý dự án .....</b>	<b>93</b>
1. Phần 1. Trang chủ .....	93
2. Phần 2. Cập nhật dự án .....	97
3. Phần 3. Lập lịch cho dự án .....	106
4. Phần 4. Hệ thống .....	118
5. Phần 5. Đăng nhập và đăng xuất .....	121

## BẢNG CÁC CHỮ VIẾT TẮT

Viết tắt	Từ tiếng Anh	Nghĩa tiếng Việt
AOA	<i>Ativities On Arcs (Network)</i>	(mạng) Các công việc là các cung
AON	<i>Ativities On Notes (Network)</i>	(mạng) Các công việc là các đỉnh
WBS	<i>Work Breakdown Structure</i>	Bảng phân rã công việc
CPM	<i>Critical Path Method</i>	Phương pháp đường găng
PERT	<i>Program Evalution and Review Technique</i>	Kỹ thuật xem xét và đánh giá chương trình (Phương pháp sơ đồ mạng)

## BẢNG DANH MỤC CÁC HÌNH

Hình 1.1. Sơ đồ tổng quát lập kế hoạch dự án .....	16
Hình 2.1: Sơ đồ tổng quát quá trình đánh giá khả thi dự án .....	20
Hình 2.1: Sơ đồ khái niệm xác định các đỉnh trung gian.....	27
Hình 2.2: Sơ đồ khái niệm vẽ mạng ban đầu .....	28
Hình 2.3: Sơ đồ khái niệm thêm công việc giả vào mạng ban đầu.....	29
Hình 2.4: Mạng công việc AOA sau khi kết thúc bước 4d.....	33
Hình 2.5: Mạng công việc AOA sau khi kết thúc bước 4.....	33
Hình 2.6: Mạng công việc AOA sau khi kết thúc bước 6.....	34
Hình 2.7: Mạng công việc với các tham số thời gian được tính toán .....	35
Hình 2.8: Biểu đồ Gantt kế hoạch lịch của ví dụ .....	36
Hình 2.9. Biểu đồ sử dụng nguồn lực (người) của ví dụ.....	36
Hình 3.1: Sơ đồ tiến trình chuyển sang lập mạng trên máy .....	40
Hình 3.2: Sơ đồ các bước xác định đỉnh trung gian.....	42
Hình 3.3: Tìm số công việc là nhỏ nhất của các dòng chưa xét.....	43
Hình 3.4: Đánh dấu các bộ công việc có CVDT là nhỏ nhất.....	44
Hình 3.5: Xóa bộ công việc đã đánh dấu có mặt trong các bộ khác .....	45
Hình 3.6. Sơ đồ vẽ mạng: xác định đầu, cuối của các công việc.....	46
Hình 3.7: Thêm đỉnh 0 và vẽ các công việc đi ra từ nó .....	47
Hình 3.8a: Thêm đỉnh trung gian <b>k</b> và vẽ công việc đi ra từ <b>k</b> .....	47
Hình 3.8b: Thêm đỉnh trung gian <b>k</b> và vẽ công việc đi ra từ <b>k</b> .....	48
Hình 3.9: Sơ đồ thuật toán vẽ đỉnh kết thúc mạng.....	49
Hình 3.10: Sơ đồ thuật toán đánh số lại các đỉnh của mạng .....	49
Hình 3.11: Sơ đồ thuật toán thêm các công việc giả.....	50
Hình 3.12: Sơ đồ logic tính thời gian bắt đầu sớm nhất của một đỉnh .....	51
Hình 3.13: Sơ đồ logic tính thời gian kết thúc muộn nhất của các đỉnh.....	52
Hình 3.14: Sơ đồ logic tính thời gian dự phòng của công việc.....	52
Hình 3.15: Sơ đồ logic vẽ biểu đồ Gantt của kế hoạch dự án.....	53
Hình 3.16: Sơ đồ logic vẽ biểu đồ sử dụng nguồn lực của dự án. ....	54
Hình 3.17: Biểu đồ Gantt kế hoạch lịch của ví dụ .....	58
Hình 3.18: Biểu đồ sử dụng nguồn lực của ví dụ.....	58
Hình 4.1: Quá trình xác định và triển khai dự án.....	60
Hình 4.2: Cấu trúc hệ thống thực đơn của chương trình.....	64
Hình 4.3: Sơ đồ phân tích xác định công việc .....	69
Hình 4.4: Biểu đồ Gantt của lịch biểu kế hoạch của bài toán – phiên bản 1.00 .....	72

Hình 4.5: Biểu đồ sử dụng nguồn lực lịch biểu của bài toán – phiên bản 1.00 .....	72
Hình 4.6: Biểu đồ Gantt của lịch biểu kế hoạch của bài toán – phiên bản 1.01 .....	75
Hình 4.7: Biểu đồ sử dụng nguồn lực của lịch biểu bài toán – phiên bản 1.01 .....	75
Hình A1.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 1 .....	82
Hình A1.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 1 .....	82
Hình A2.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 2 .....	85
Hình A2.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 1 .....	85
Hình A3.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 3 .....	88
Hình A3.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 3 .....	88
Hình A4.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 4 .....	91
Hình A4.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 4 .....	92

## BẢNG DANH MỤC CÁC BẢNG

Bảng 2.1: Bảng phân rã công việc .....	30
Bảng 2.2: Thực hiện các bước của giai đoạn 1 .....	31
Bảng 2.3: Kết quả thực hiện các bước của giai đoạn 1 .....	32
Bảng 2.4: Bảng công việc còn lại sau 4 lần lặp lại bước 4 của giai đoạn 2.....	32
Bảng 3.1. Bảng tính toán hệ số hoàn vốn (phân tích chi phí - hiệu quả).....	38
Bảng 3.2. Bảng cấu trúc dữ liệu cho bài toán lập kế hoạch lịch trên máy.....	41
Bảng 3.3. Bảng phân rã chức năng công việc (đầu vào).....	56
Bảng 3.4. Bảng xác định các đỉnh trung gian .....	56
Bảng 3.5. Bảng tính các tham số thời gian các đỉnh của mạng .....	57
Bảng 3.6. Bảng lịch thực hiện công việc và đường găng.....	57
Bảng 3.7. Tổng hợp các kết quả thử nghiệm chương trình thuật toán.....	59
Bảng 4.0. Bảng phân rã công việc của bài toán .....	69
Bảng 4.1. Bảng phân rã chức năng của bài toán - phiên bản 1.00 .....	70
Bảng 4.3. Bảng thời gian bắt đầu, kết thúc của các đỉnh-phiên bản 1.00 .....	70
Bảng 4.4. Bảng lịch thực hiện công việc của bài toán-phiên bản 1.00 .....	71
Bảng 4.5. Bảng phân rã chức năng của bài toán - phiên bản 1.01 .....	73
Bảng 4.6. Bảng tính toán trung gian của bài toán - phiên bản 1.01 .....	73
Bảng 4.7. Bảng thời gian bắt đầu, kết thúc của các đỉnh-phiên bản 1.01 .....	74
Bảng 4.8. Bảng lịch thực hiện công việc -phiên bản 1.01 .....	74
Bảng A1.1. Bảng dữ liệu đầu vào ví dụ 1 .....	80
Bảng A1.2. Kết quả trung gian ví dụ 1 .....	80
Bảng A1.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 1.....	81
Bảng A1.4. Bảng lịch thời gian thực hiện công việc của ví dụ 1.....	81
Bảng A2.1. Bảng dữ liệu đầu vào ví dụ 2 .....	83
Bảng A2.2. Kết quả trung gian ví dụ 2 .....	83
Bảng A2.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 2.....	84
Bảng A2.4. Bảng lịch thời gian thực hiện công việc của ví dụ 2.....	84
Bảng A3.1. Bảng dữ liệu đầu vào ví dụ 3 .....	86
Bảng A3.2. Kết quả trung gian ví dụ 3 .....	86
Bảng A3.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 3.....	87
Bảng A3.4. Bảng lịch thời gian thực hiện công việc của ví dụ 3.....	87
Bảng A4.1. Bảng dữ liệu đầu vào ví dụ 4 .....	89
Bảng A4.2. Kết quả trung gian ví dụ 4 .....	90
Bảng A4.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 4.....	90
Bảng A4.4. Bảng lịch thời gian thực hiện công việc của ví dụ 4.....	91



## LỜI CẢM ƠN

Trân trọng cảm ơn tất cả các Giáo sư, Phó giáo sư, Tiến sĩ, các thầy giáo cô giáo của Khoa Công nghệ Thông tin trường Đại Học Dân Lập Hải Phòng đã nhiệt tình giảng dạy, tạo điều kiện thuận lợi cho tác giả trong quá trình học tập, nghiên cứu, hoàn thành chương trình học tập của khóa học.

Tác giả xin trân trọng cảm ơn PGS.TS. Nguyễn Văn Vy, các thầy cô trong hội đồng khoa học đã giành thời gian chỉ bảo tận tình giúp em hoàn thành luận văn.

Tác giả xin chân thành cảm ơn Sở GD&ĐT Hải Phòng, Ban giám hiệu và giáo viên trường THPT Nguyễn Khuyến huyện Vĩnh Bảo thành phố Hải Phòng đã quan tâm giúp đỡ tạo mọi điều kiện thuận lợi cho tác giả trong suốt quá trình học tập, nghiên cứu và hoàn thành luận văn.

Tác giả xin cảm ơn gia đình, bạn, đồng nghiệp, đã động viên tiếp thêm nghị lực để tác giả hoàn thành khóa học và luận văn.

Mặc dù đã có nhiều cố gắng, song luận văn khó tránh khỏi những thiếu sót. Tác giả rất mong sự chỉ bảo, góp ý của các nhà khoa học, các thầy cô giáo và đồng nghiệp.

Xin chân trọng cảm ơn!

Hải Phòng, ngày 01 tháng 12 năm 2016

Tác giả

Trần Nhật Giáp

## LỜI CAM ĐOAN

Tôi xin cam đoan rằng, đây là công trình nghiên cứu của tôi trong đó có sự giúp đỡ rất lớn của thầy *PGS.TS. Nguyễn Văn Vỵ*. Các nội dung nghiên cứu và kết quả trong đề tài này là hoàn toàn trung thực.

Trong luận văn, tôi có tham khảo đến một số tài liệu của một số tác giả đã được liệt kê tại phần Tài liệu tham khảo ở cuối luận văn.

*Hải Phòng, ngày 01 tháng 12 năm 2016*

**Tác giả**

Trần Nhật Giáp

# MỞ ĐẦU

## 1. Đặt vấn đề

Tại Việt Nam, trong nhiều năm gần đây, khái niệm dự án trở nên quen thuộc đối với mọi người, đặc biệt là các nhà kinh doanh và người quản lý các cấp. Nhiều hoạt động trong các cơ quan, tổ chức, doanh nghiệp được thực hiện theo hình thức dự án. Hai nhiệm vụ cực kỳ quan trọng trong việc thực hiện một dự án đó là:

- a. *Đánh giá dự án.* Trước khi triển khai một dự án, ta cần phải biết dự án có đáng đầu tư hay không. Công việc này cần tiến hành ở giai đoạn đầu tiên: Giai đoạn lựa chọn dự án. Bất kỳ một người nào có chút hiểu biết về dự án thì đều có nhu cầu và không thể bỏ qua được công việc này.
- b. *Lập lịch để thực hiện dự án.* Đây là công việc đầu tiên của giai đoạn thực hiện dự án. Thực chất, đây là công việc lập kế hoạch cho quản lý dự án. Sau lần lập đầu tiên, công việc này thường phải lặp lại nhiều lần sau mỗi chu kỳ thời gian nhất định, để lập kế hoạch cho giai đoạn tiếp theo. Do phải tiến hành nhiều lần trong một dự án, vì vậy có nhu cầu cấp thiết phải tự động hóa hoạt động này để giảm công sức của người quản lý, và đặc biệt quan trọng hơn cả là để đáp ứng kịp thời hoạt động quản lý mà dự án yêu cầu trong từng giai đoạn quản lý.

Trên thực tế, nhiều người vẫn thực hiện các công việc này bằng cách thủ công với sự trợ giúp của máy tính. Cách đó thường mất nhiều thời gian, và không có điều kiện để lựa chọn một dự án hay một phương án tốt từ nhiều lựa chọn có thể. Mặc dù đã có những chương trình phần mềm quản lý dự án, nhưng chỉ những cơ quan chuyên nghiệp hay các nhà chuyên môn mới biết sử dụng nó thành thạo.

Để cho nhiều người, trong những hoàn cảnh khác nhau đều có thể thực hiện hai công việc trên một cách dễ dàng và thuận tiện, phục vụ cho công việc của mình, cần thiết phải xây dựng chương trình để tự động hóa hai hoạt động này và cài đặt trên máy tính giúp người bình thường có thể sử dụng được ngay. Với mục đích đó, đề tài “Đánh giá dự án đầu tư và lập lịch cho quản lý dự án tự động” được chọn làm đề tài luận văn cao học. Mục tiêu của luận văn là tạo ra một bộ công cụ trên

máy tính để trợ giúp cho những người làm dự án có thể thực hiện hai nhiệm vụ trên khi có yêu cầu. Với bộ công cụ này, người bình thường biết dùng máy tính đều có thể sử dụng được ngay, không đòi hỏi phải đào tạo nhiều thời gian như các bộ chương trình quản lý dự án chuyên nghiệp.

## **2. Đối tượng và phạm vi nghiên cứu**

### **2.1. Đối tượng nghiên cứu**

Đối tượng nghiên cứu của luận văn bao gồm:

- Công cụ đánh giá sự khả thi của một dự án đầu tư
- Công cụ lập lịch thực hiện dự án bằng cách sử dụng mạng công việc với mỗi cung là một công việc (Activity On Arc - AOA), mà đã có thuật toán vẽ mạng bằng tay được nghiên cứu [10].

### **2.2. Phạm vi nghiên cứu**

Luận văn không nghiên cứu toàn bộ quá trình quản lý dự án, mà chỉ tập trung nghiên cứu các nội dung chính sau:

- Trong việc phân tích dự án khả thi, sau khi đã xác định được các yếu tố chi phí, chỉ tiến hành tính toán hai chỉ tiêu quan trọng là hệ số hoàn vốn và *thời gian thu hồi vốn* của dự án để làm cơ sở lựa chọn dự án.
- Trong việc lập lịch dự án, chỉ xây dựng một *lịch biểu dự án về thời gian*: bao gồm thời gian bắt đầu và kết thúc để thực hiện mỗi công việc, cùng với thời gian dự phòng của nó, và tổng thời gian để thực hiện toàn bộ dự án. Khi xây dựng lịch này, chưa xét đến các yếu tố hạn chế về nguồn lực (tài nguyên) khác nhau cần thiết cho việc thực hiện dự án. Bài toán lập lịch với các ràng buộc về tài nguyên là một bài toán khó, không thể giải quyết trong phạm vi một luận án thạc sĩ.
- Tạo một bộ công cụ cho phép tự động hóa lập kế hoạch dự án đáp ứng được yêu cầu thay đổi về cấu trúc dự án (các công việc và trình tự) cũng như thay đổi những dữ liệu về thời gian và nguồn lực dùng cho công việc trong quá

trình thực hiện dự án. Đây là những sự kiện thường xuyên xảy ra trong hoạt động quản lý dự án. Nếu không có công cụ thích hợp sẽ không quản lý được dự án hoặc đưa đến những thiệt hại cho dự án

### **3. Hướng nghiên cứu của đề tài**

Sử dụng phương pháp mô hình hóa để lập mô hình bài toán, và phân tích thiết kế để xây dựng chương trình tự động hóa giải quyết bài toán đặt ra trên máy tính.

### **4. Những nội dung nghiên cứu chính**

- Phân tích các yếu tố tham gia vào quá trình đánh giá và lập lịch dự án và tiến hành mô hình hóa cho mỗi quá trình này.
- Tiến hành thiết kế cấu trúc dữ liệu cho mỗi bài toán và thiết kế quy trình giải mỗi bài toán theo mô hình lập ra.
- Tiến hành xây dựng chương trình, cài đặt, thử nghiệm chúng với những ví dụ bằng số cho hai bài toán này và đánh giá tính khả dụng của các chương trình.

### **5. Phương pháp nghiên cứu**

- Sử dụng phương pháp mô hình hóa để lập mô hình các bài toán đặt ra [4].
- Áp dụng phương pháp phân tích thiết kế hướng cấu trúc để phát triển các bộ công cụ chương trình cho việc giải các bài toán đặt ra.

### **6. Ý nghĩa khoa học của đề tài**

Chuyển các công việc làm tay sang tự động hóa trên máy tính là một nhu cầu rất lớn trong đời sống xã hội. Luận văn tạo ra bộ công cụ giúp cho hoạt động lựa chọn dự án và lập lịch dự án thực hiện nhanh chóng, mất ít công sức, lựa chọn nhiều phương án khác nhau để đạt hiệu quả cao, đáp ứng được yêu cầu của thực tiễn của quản lý dự án.

# CHƯƠNG I: THỰC TRẠNG VỀ QUẢN LÝ DỰ ÁN VÀ LẬP LỊCH TRONG QUẢN LÝ DỰ ÁN

## 1.1. Khái quát về quản lý dự án

### 1.1.1. Định nghĩa dự án

*Dự án là một nhiệm vụ cần hoàn thành để tạo ra một sản phẩm hay dịch vụ duy nhất, trong một thời hạn đã cho, với kinh phí dự kiến [11]. Theo Viện Quản lý dự án Mỹ [24] thì: “Dự án là một sự cố gắng nhất thời được tiến hành để tạo ra một sản phẩm hay dịch vụ”.*

Như vậy, dự án có một số đặc trưng sau:

- Sản phẩm dự án là duy nhất (không có cái thứ hai đồng nhất với nó).
- Thực hiện trong một thời gian nhất định, không lặp lại và cũng không được kéo dài quá hạn.
- Bị giới hạn trước về thời gian và kinh phí (được xác định trước).

Với những đặc trưng trên đây, việc thực hiện dự án đòi hỏi phải tuân thủ các yêu cầu rất khắt khe nhằm đảm bảo có được chất lượng sản phẩm/dịch vụ và thỏa mãn cả yêu cầu cả về chi phí và thời gian.

Một trong các yếu tố đảm bảo sự thành công của dự án [5], [11] là có bản kế hoạch tốt cho dự án và có các công cụ hữu hiệu để phục vụ công tác quản lý dự án. Vì vậy, trong luận văn này tập trung giải quyết hai trong số những nhiệm vụ quan trọng của quản lý dự án là:

- Đánh giá dự án xem có đáng để thực hiện hay không,
- Thực hiện việc lập kế hoạch cho quá trình quản lý sao cho đáp được yêu cầu kịp thời và đảm bảo chất lượng dự án là cơ sở cho việc thực hiện dự án thành công.

### 1.1.2. Đánh giá khái quát để lựa chọn dự án

**Khi có một dự án, chúng ta cần phải đánh giá, xem xét xem dự án có đáng thực hiện không và khả năng thực hiện dự án có đảm bảo xác suất thành công cao hay không. Một trong các điều kiện này không thỏa mãn thì dự án không được thực hiện.**

Mục tiêu cơ bản của đánh giá dự án, nói một cách khác là khẳng định sự cần thiết và sự khả thi của nó. Luận chứng tính khả thi của dự án thường được xem xét trên các mặt sau đây:

- *Khả thi về kinh tế* (đủ vốn, thu hồi được vốn và có lãi)
- *Khả thi về công nghệ* (có công nghệ phù hợp sẵn sàng, đối với các dự án liên quan đến công nghệ)
- *Khả thi về thực hiện* (đủ năng lực và các nguồn lực để thực hiện)
- *Khả thi về thời gian* (có thể hoàn thành dự án trong thời gian cho phép).

Vì dự án là một khoản đầu tư, mọi hoạt động đầu tư luôn hướng đến hiệu quả kinh tế. Vì thế khả thi về kinh tế là một tiêu chí quan trọng cho mọi dự án và thường được xem xét đầu tiên. Trong luận văn này chỉ tập trung nghiên cứu về khả thi kinh tế.

### 1.1.3. Đánh giá khả thi kinh tế của dự án

Để đánh giá khả thi về kinh tế, đầu tiên ta phải xác định xem tổng chi phí cho dự án là bao nhiêu, có nằm trong phạm vi tổng vốn ban đầu dành cho dự án không. Như chúng ta đã biết, mỗi dự án là duy nhất theo nghĩa không có cái nào giống nó (do làm lần đầu tiên hoặc hoàn toàn mới). Do đó cần có cách thức để tìm ra các khoản mục chi phí cần thiết của dự án và ước lượng chi phí cho các khoản mục này. Có nhiều phương pháp cho phép xác định các khoản chi phí đầu tư cũng như ước lượng chi phí cho nó đã được các tài liệu giảng dạy và hướng dẫn về quản lý giới thiệu, chúng ta sẽ không bàn luận ở đây. Nhưng có một vấn đề cần nói đến là: để đạt đến cùng một sản phẩm của dự án thường có nhiều phương án thực hiện khác nhau. Và với mỗi phương án ấy, thì các khoản mục chi phí cũng như số lượng chi phí theo mỗi khoản mục sẽ khác nhau. Do đó có nhu cầu tính toán chi phí cũng như

lợi nhuận thu được theo nhiều phương án để lựa chọn phương án thích hợp cho dự án.

Đánh giá khả thi trong luận văn sẽ tập trung vào việc tính toán tổng chi phí cho từng phương án của dự án (để xem có nằm trong phạm vi vốn cho phép không) và khả năng thu hồi vốn (hiệu quả đầu tư) trong mỗi trường hợp quyết định lựa chọn dự án.

## **1.2. Lập kế hoạch dự án và bài toán lập lịch**

### **1.2.1. Sơ đồ tổng thể lập kế hoạch dự án**

Sơ đồ tổng quát của việc lập kế hoạch dự án cho ở hình 1.1.

Trong sơ đồ, các công đoạn 2, 4, 5 là quan trọng nhất, vì nó tạo ra kế hoạch lịch biểu của dự án. Các kế hoạch khác của dự án được thiết lập dựa trên kế hoạch lịch.

Để thực hiện việc lập kế hoạch trên, người ta đã sử dụng hai phương pháp:

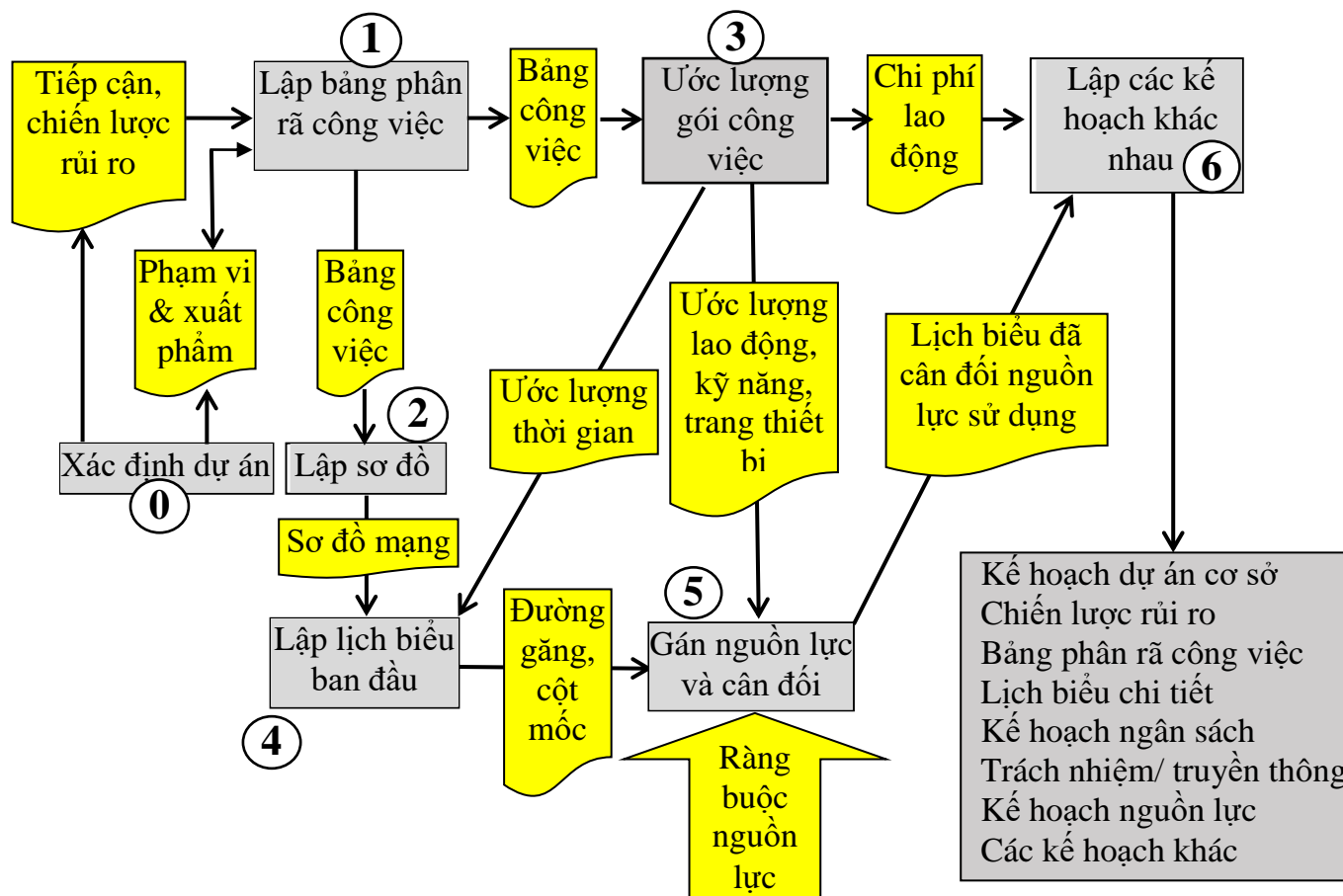
- Phương pháp đường găng (Critical Path Method – CPM)[20].
- Phương pháp sơ đồ mạng PERT (Program Evaluation and Review Technique) [11]

Từ đó đến nay, hai phương pháp này được ứng dụng rộng rãi trong các tổ chức, đặc biệt trong lĩnh vực quản lý dự án ([2], [3], [5], [14], [15], [16], [19], [24], ...).

Trong quy trình lập kế hoạch, lập sơ đồ mạng (2) là khâu mô hình hóa quan trọng, làm cơ sở cho việc tính toán và thực hiện các bước tiếp theo, mà nội dung về cơ bản là như nhau. Ở bước này, mỗi phương pháp lại sử dụng một dạng sơ đồ mạng khác nhau.

CPM sử dụng sơ đồ mạng dạng AON (Activities On Nodes), mà mỗi đỉnh biểu diễn một công việc, mỗi cung nối hai công việc với nhau biểu diễn mối quan hệ trình tự giữa chúng. Cách lập sơ đồ dạng này rất đơn giản, ai cũng có thể lập được một cách dễ dàng. Nhưng việc ứng dụng mạng dạng AON chỉ được sử dụng trong một phạm vi hạn chế.





Hình 1.1. Sơ đồ tổng quát lập kế hoạch dự án

Phương pháp PERT sử dụng sơ đồ mạng dạng AOA (Activities On Arcs), trong đó mỗi cung biểu diễn một công việc, mỗi đỉnh biểu diễn sự kết thúc của một số công việc và là điểm bắt đầu cho một số công việc khác. Việc lập mạng AOA thường không đơn giản vì phải sử dụng đến các đỉnh giả và công việc giả. Số lượng các công việc giả sẽ đặc biệt nhiều nếu dự án có quy mô lớn. Khi đó, với cùng một bảng công việc đã cho, người ta có thể đưa ra một vài mạng AOA khác nhau.

Trong nhiều tài liệu trình bày về phương pháp PERT ([1], [6], [12], [27]) cũng như trong các tài liệu ứng dụng nó, để lập mạng AOA các tác giả chỉ đưa ra những quy tắc hướng dẫn mang tính kinh nghiệm và chỉ ra một số trường hợp cụ thể vẽ mạng AOA, để giải thích cách vẽ như thế nào là đúng, là sai.

Mạng AOA có nhiều lợi thế: Vì là một đồ thị có hướng, nó có thể được chuyển sang dạng ma trận để tính toán trên một bảng. Nó cũng rất thích hợp cho các kỹ thuật phân tích và mô hình hóa nhiều bài toán tối ưu ([12], [6]) như bài toán luồng cực đại, tối ưu hóa phân phối nguồn lực của dự án,...

Vì vai trò quan trọng của mạng AOA, nên có nhiều phương pháp lập mạng AOA đã được nghiên cứu. Một loạt các phương pháp trực cảm để lập mạng AOA hướng đến việc tối thiểu hóa các công việc giả ([14], [18], [19], [22], [25], [26]) đã được đề xuất. Các phương pháp đi theo hướng này gặp nhiều khó khăn, vì theo Krishnamoorty và Deon [21] bài toán lập mạng AOA tối thiểu hóa các cung giả là bài toán khó.

Mouhoub và Benhocine [23] đã trình bày một thuật toán để xây dựng mạng AOA bằng cách chuyển một mạng AON sang mạng AOA. Thuật toán là khá phức tạp, và theo các tác giả, độ phức tạp tính toán của thuật toán là  $O(n^4)$ . Một thuật toán khác xây dựng mạng AOA được Yuval Cohen, Arik Sadeh [12] đề xuất, cho phép xác định duy nhất mạng AOA từ một WBS (Work Breakdown Structure) đã cho. Tuy nhiên thuật toán chỉ được xem xét và thẩm định qua các ví dụ mà không được chứng minh đầy đủ.

Một thuật toán mới cho phép lập một mạng AOA duy nhất, đơn giản và dễ dàng từ một WBS cho trước. Thuật toán này đã được giới thiệu trong tài liệu [9] của

các tác giả trước đây, nhưng gần đây mới đưa ra các chứng minh đầy đủ về tính đúng đắn, tính hữu hạn và tính duy nhất nghiêm của thuật toán[10].

### **1.2.2. Các khó khăn của việc lập kế hoạch dự án và bài toán lập lịch**

Việc lập kế hoạch dự án thường diễn ra nhiều lần trong quá trình quản lý dự án. Nó không chỉ cần đến khi bắt đầu một giai đoạn mới của quá trình quản lý dự án, mà còn phải thực hiện khi có những thay đổi xảy ra đối với sản phẩm, đối với tiến độ bàn giao và đối với tính chất của một số công việc; mà những sự kiện này lại rất thường hay xảy ra trong quản lý dự án. Vì thế, để đáp ứng yêu cầu cấp bách của quản lý dự án, nhiều chương trình phần mềm được xây dựng giúp tự động hóa hoạt động này. Đối với những chương trình dựa trên việc lập mạng AON không có khó khăn gì đặt ra. Tuy nhiên, những chương trình loại này đã không cho phép giải quyết triệt để bài toán lập lịch trong trường hợp nguồn lực hạn chế (giai đoạn 5 trong sơ đồ hình 1.1.)

Nhưng với những chương trình dựa trên mạng AOA, hiện nay người ta phải lập mạng bằng tay, sau đó nhập dữ liệu mạng này vào chương trình để thực hiện các bước tiếp. Điều đó có nghĩa là, chương trình chỉ thực hiện ở mức bán tự động, mà không được tự động hóa hoàn toàn. Nhờ có thuật toán mới, mặc dù được mô tả cho việc vẽ tay, nhưng do tính logic chặt chẽ của thuật toán, nên có thể chuyển nó sang chương trình để tự động hóa việc lập mạng. Và nhờ vậy có thể tự động hóa hoàn toàn bài toán lập lịch. Hơn nữa, việc sử dụng AOA sẽ hỗ trợ giải quyết triệt để bài toán lập lịch có tính đến các ràng buộc về nguồn lực như một số tài liệu đã đề cập đến ([2], [6], [12]).

### **1.2.3. Một số phần mềm đã sử dụng để lập lịch**

Do những khó khăn của quản lý dự án nói chung và bài toán lập lịch nói riêng, đã có nhiều các phần mềm được xây dựng với mục đích có thể xử lý hiệu quả và dễ sử dụng đối với bài toán lập lịch biểu này.

Một trong số đó phải kể đến phần mềm quản lý MS (MicrosoftProject), là một phần mềm quản lý dự án được Microsoft phát triển [5]. Chương trình này được thiết kế quản lý hỗ trợ dự án, theo dõi tiến độ, quản lý ngân sách và phân tích khối lượng

công việc. Phần mềm này được thiết kế dựa trên thuật toán của phương pháp lập tiến độ theo sơ đồ mạng CMP và sơ đồ Gantt.

Ưu điểm của MS Project là một phần mềm linh hoạt, người sử dụng không bị gò bó theo trình tự các bước cứng nhắc trong việc lập và xác định các quá trình thực hiện dự án. Người sử dụng có thể bắt đầu từ việc thao tác vài ý tưởng về công tác cần thiết nhất cho dự án. Ngoài ra người sử dụng còn có thể điều chỉnh lịch trình về sau, cập nhật liên tục thông tin của dự án, rà soát việc tính toán và tùy chọn hiển thị. Việc sử dụng MS Project sẽ giúp cho người quản lý dự án trong các công việc sau:

- Tổ chức lập kế hoạch và quản lý tổng thể.
- Lên lịch thực hiện các công việc.
- Điều chỉnh lại các kế hoạch thực hiện dự án cho phù hợp với các điều kiện ràng buộc.

Nhược điểm của MS Project: Mặc dù có rất nhiều các ưu điểm và tính hữu dụng, tuy nhiên, MS Project cũng có một số nhược điểm. Là một phần mềm được xây dựng dựa trên mạng CMP (mỗi công việc sẽ là một nút của mạng) nên việc cân đối nguồn lực trong các công việc có các ràng buộc gặp nhiều khó khăn. Quá trình cân đối nguồn lực chủ yếu dựa vào làm tay và dựa trên kinh nghiệm của các nhà quản lý để di chuyển các công việc có thời gian dự phòng. Như vậy sẽ không thể cân đối nguồn lực đối với trường hợp dữ liệu lớn và các ràng buộc phức tạp và không thể tự động hóa hoàn toàn quá trình cân đối nguồn lực.

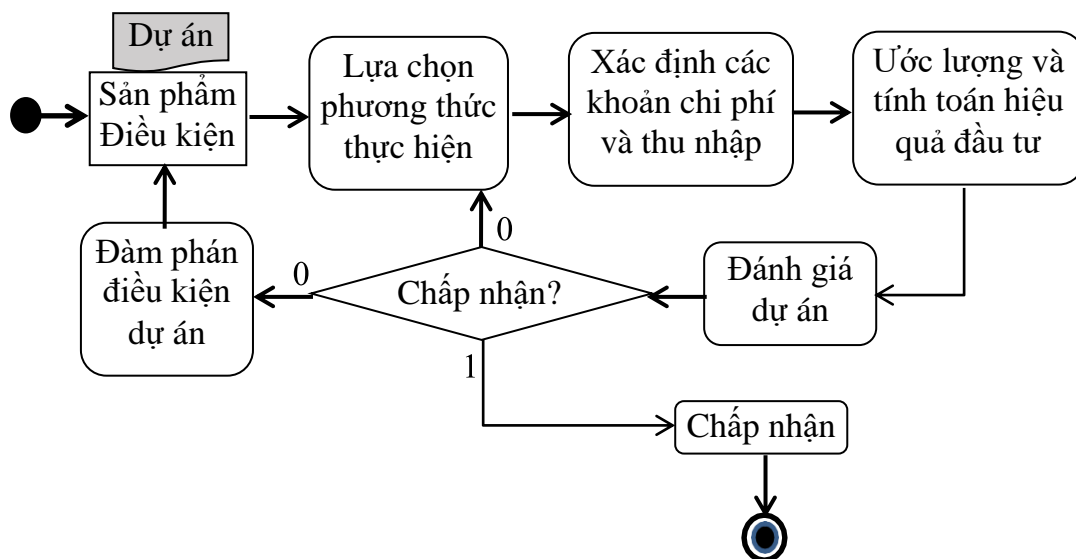
Đối với việc xây dựng bài toán lập lịch dưới đây sẽ giúp cho việc thực hiện quản lý dự án, lập lịch biểu, cân đối nguồn lực đối với các công việc có ràng buộc sẽ được tự động hóa hoàn toàn.

## CHƯƠNG II: ĐÁNH GIÁ KHẢ THI VÀ LẬP KẾ HOẠCH LỊCH THỜI GIAN BẰNG TAY

### 2.1. Đánh giá khả thi kinh tế của dự án

#### 2.1.1. Sơ đồ thực hiện đánh giá khả thi kinh tế của dự án

Quá trình đánh giá khả thi kinh tế một dự án có thể mô tả như ở sơ đồ hình 2.1.



Hình 2.1: Sơ đồ tổng quát quá trình đánh giá khả thi dự án

#### a. Lựa chọn phương thức thực hiện

Với mỗi sản phẩm hay dịch vụ, ta có nhiều cách khác nhau (phương thức) để thực hiện chúng. Chẳng hạn có thể làm hoàn toàn thủ công hay tự động hóa một phần. Những người có kinh nghiệm, trải qua thực tiễn thường có thể đề xuất hay đưa ra nhiều phương thức khác nhau để thực hiện đối với một dự án đã cho. Ví dụ đơn giản, để mua một sản phẩm hiện nay, ta có thể trực tiếp đến một cửa hàng để mua, nếu ở gần cửa hàng. Nếu ở xa, ta phải đi phương tiện giao thông đến đó. Ta cũng có thể đặt hàng qua mạng, nếu ta có nối mạng. Càng nhiều phương thức, ta

càng có nhiều khả năng để lựa chọn được phương án thích hợp sau khi tính toán được chi phí của chúng.

### ***b. Xác định các khoản chi phí***

Với mỗi phương thức thực hiện, ta phải xác định các khoản chi phí tương ứng để có được sản phẩm/dịch vụ của dự án. Thông thường, bằng cách phân tích các yếu tố tham gia vào việc tạo ra sản phẩm đối với mỗi phương thức. Từ đó ta sẽ chỉ ra dần các khoản mục chi tiết hơn cần chi phí cho dự án. Chẳng hạn như ví dụ ở trên, nếu ta đặt hàng qua mạng, thì các khoản chi phí sẽ là: *tiền chi phí dùng mạng, tiền trả cho hàng mua và tiền chi phí vận chuyển hàng đến nhà.*

### ***c. Ước lượng và tính toán chi phí***

Việc ước lượng và tính toán các khoản chi phí cũng như thu nhập tùy vào những trường hợp cụ thể, thường dựa vào giá cả hiện hành. Nếu các khoản mục là quá gộp thì có thể phân nhỏ thành các yếu tố chi phí thành phần để có thể áp giá có sẵn trên thực tế. Việc tính toán chi phí và hiệu quả sẽ trình bày ở mục sau.

### ***d. Đánh giá khả thi của dự án***

Đánh giá đầu tiên là xem tổng chi phí để đạt được sản phẩm dự án có vượt quá số vốn ban đầu dự án đặt ra không, vượt nhiều hay ít? Nếu với các phương án khác nhau đều có tổng chi phí vượt qua số vốn dự án đủ lớn thì cần đàm phán với chủ dự án để tăng tiền vốn hay giảm yêu cầu về sản phẩm. Trong trường hợp ngược lại, có thể thay đổi các yếu tố để giảm chi phí nhằm đạt được yêu cầu của dự án. Tiếp theo, là tính toán khả năng hoàn vốn và lợi nhuận thu được. Rõ ràng thời gian hoàn vốn dài hoặc không có lãi thì cũng khó có thể chấp nhận được dự án.

## **2.1.2. Tính toán hệ số hoàn vốn và thời gian hoàn vốn**

### ***a. Các yếu tố thành phần để tính hệ số hoàn vốn***

Các yếu tố để tính hệ số hoàn vốn chia thành hai loại: *chi phí* và *doanh thu*.

Trong chi phí, người ta lại chia làm hai loại: *chi phí ban đầu* (một lần) và *chi phí thường xuyên* (theo thời gian). *Chi phí ban đầu* thường là các khoản đầu tư cho cơ sở vật chất và trang thiết bị để phục vụ làm ra sản phẩm/dịch vụ của dự án (như

nhà cửa, trang thiết bị, phương tiện kinh doanh,...). Chi phí này thực hiện một lần, vì những “tài sản” thường tồn tại một thời gian đủ dài so với một chu kỳ làm ra sản phẩm/dịch vụ của dự án. Chi phí thường xuyên là những chi phí về nguyên vật liệu, năng lượng... và nhân công để trực tiếp làm ra sản phẩm. Thường những chi phí này tỷ lệ với số sản phẩm/dịch vụ làm ra.

*Doanh thu* là các khoản thu được do tiêu thụ sản phẩm/dịch vụ và các khoản thu khác đi kèm, và có thể còn chưa rõ ràng. Vì thế, người ta thường chỉ tính những khoản thu trực tiếp từ sản phẩm có thể tính được.

### ***b. Nguyên tắc tính hệ số hoàn vốn***

Như trên đã thấy, các khoản chi phí một lần khác với các khoản chi thường xuyên và doanh thu về thời điểm phát sinh: Chi phí một lần thường bỏ ra khi chưa có sản phẩm. Còn chi thường xuyên và doanh thu nhận được ở những thời điểm khác nhau trong tương lai sau này khi dự án cho ra sản phẩm. Rõ ràng, về mặt giá trị, chúng không thể so sánh được với nhau. Vì thế, người ta phải quy đổi chúng về cùng thời điểm để giá trị của chúng có thể so sánh được. Ta có thể quy giá trị của tất cả chi phí thường xuyên và doanh thu về thời gian hiện tại (khi đầu tư) hay vào lúc kết thúc dự án. Thông thường, người ta quy những giá trị phát sinh thường xuyên về thời điểm hiện tại, lúc bắt đầu dự án.

Giả sử ta có một số tiền  $A$  đem gửi ngân hàng với lãi suất  $r$  % sau một đơn vị thời gian (tháng, quý, năm). Như vậy số tiền thu được ở thời kỳ  $k$  sẽ là:

$$B(\text{số tiền thu được ở thời kỳ } k) = A \cdot (1+r)^k$$

Từ đó ta suy ra: Nếu ta có số tiền bằng  $B$  ở kỳ thứ  $k$  sau này, khi quy đổi nó về thời điểm hiện tại, ta nhận được giá trị:

$$A = B / (1+r)^k \quad (1)$$

Hệ số  $1/(1+r)^k$  được gọi là **hệ số quy đổi** của số tiền  $B$  nhận được ở thời kỳ thứ  $k$  về giá trị hiện tại (*present value*: PV) với lãi suất  $r$  % (cho một chu kỳ).

Như vậy, để tính toán về hiệu quả đầu tư, ta phải quy giá trị của các chi phí thường xuyên cũng như doanh thu trong các kỳ sau này về thời kỳ hiện tại (thời điểm lấy làm gốc) theo công thức (1)

### c. Công thức tính hệ số hoàn vốn

Giả sử ta có một dự án với đầu tư ban đầu (tổng vốn) là **TV**, và có lãi suất vay vốn là **r** thì:

- Doanh thu ở kỳ thứ *i* là  $DT_i$ , quy đổi về giá trị hiện tại ta được:  $DT_i/(1+r)^k$
- Chi phí cho kỳ thứ *i* là  $CF_i$ , quy đổi về giá trị hiện tại ta được:  $CF_i/(1+r)^k$
- Lợi nhuận thu được ở kỳ thứ *i* tính theo giá quy đổi sẽ là:

$$\begin{aligned} & \text{(Doanh thu kỳ } i) - \text{(Chi phí kỳ } i) \text{ hay} \\ & (DT_i/(1+r)^k - CF_i/(1+r)^k) = (DT_i - CF_i)/(1+r)^k \end{aligned}$$

Tổng lợi nhuận tính theo giá trị hiện tại trong **n** chu kỳ sẽ là:

$$LN_{pv} = (DT_1 - CF_1)/(1+r) + (DT_2 - CF_2)/(1+r)^2 + \dots + (DT_n - CF_n)/(1+r)^n$$

Công thức tính hệ số hoàn vốn **ROI** (*Return On Investemen*) được tính như sau:

$$\text{Hệ số hoàn vốn} = (\text{lợi nhuận bình quân trong kỳ}) / \text{tổng đầu tư}$$

$$\text{Hay } ROI = (LN_{pv} / n) / TV = LN_{pv} / (n \cdot TV) \quad (2)$$

Hệ số này có nghĩa là, sau **n** chu kỳ đầu tư, một đồng vốn đầu tư ban đầu sẽ đem lại **ROI** đồng lợi nhuận. Nếu **ROI** càng lớn, thì lợi nhuận càng cao. Vì vậy nó được dùng để so sánh với các khoản đầu tư khác (phương thức khác, dự án khác) để làm cơ sở quyết định lựa chọn dự án.

Và con số ngược lại của ROI: **T = 1/ROI** (3)

cho ta thời gian hoàn vốn. Nó cũng là một chỉ tiêu để đánh giá hiệu quả của một dự án đầu tư.

## 2.2. Lập kế hoạch lịch thời gian cho dự án

Như chương I đã trình bày, khâu khó khăn nhất trong quy trình lập kế hoạch dự án là khâu lập mạng công việc (2) khi sử dụng mạng AOA, mà các phần mềm vẫn phải sử dụng kết quả vẽ thủ công. Dưới đây sẽ trình bày thuật toán về lập mạng



AOA bằng tay, đã được giới thiệu trong tài liệu [9]. Nó làm cơ sở cho việc tự động hóa công đoạn này.

### 2.2.1. Thuật toán lập mạng AOA bằng tay

#### **Nội dung thuật toán:**

Đầu vào cho thuật toán là bảng phân rã công việc (WBS - *Work Breakdown Structure*) có dữ liệu được cho ở các cột: *mã công việc*, *tên công việc* và *công việc đi trước*. Kết quả cho ra là một mạng công việc dạng AOA cho phép ta tiến hành các tính toán các dữ liệu của bảng kế hoạch lịch để phục vụ quản lý dự án.

Thuật toán bao gồm các giai đoạn sau đây:

#### **a. Giai đoạn 1: Đánh dấu xác định các đỉnh trung gian**

**Bước 1: Đánh dấu các bộ công việc:** Duyệt cột “*Công việc đi trước*” của WBS. Nếu mọi bộ công việc (có thể bao gồm một hay một số công việc) trên mỗi dòng đều đã được đánh dấu (hay đưa vào trong ngoặc) hoặc bị xóa thì chuyển đến bước 3 (giai đoạn 2).

Ngược lại, mỗi lần duyệt cột này từ trên xuống, ta chỉ đánh dấu các bộ công việc còn tự do (chưa đánh dấu hoặc chưa xóa) ở trên một dòng và có số lượng công việc là ít nhất (có thể bằng 1, hoặc 2, hoặc 3,..). Như vậy, sẽ có ít nhất một bộ công việc ở trên một dòng nào đó được đánh dấu trong bước này và số dòng có bộ các công việc còn tự do trong cột này của WBS giảm đi ít nhất một dòng.

**Mỗi bộ công việc** được đánh dấu sẽ xác định **một đỉnh trung gian** của sơ đồ mạng công việc. Đỉnh trung gian này là *sự kiện kết thúc của các công việc thuộc bộ này*, và là *sự kiện bắt đầu của tất cả các công việc đi ra khỏi nó*, được xác định là những công việc có **bộ công việc này** nằm ở cột “*Công việc đi trước*” cùng dòng với nó.

**Bước 2: Xóa các bộ công việc đã được đánh dấu:** Duyệt tất cả các bộ công việc ở mỗi dòng thuộc cột “*Công việc đi trước*” của WBS còn tự do (chưa đánh dấu hoặc chưa bị xóa). Nếu chúng có chứa **bộ công việc vừa được đánh dấu** ở bước 1 thì xóa đi và quay lại bước 1.

Giai đoạn 1 kết thúc khi cột “Công việc đi trước” của WBS không còn bộ công việc tự do nào.

### ***b. Giai đoạn 2: Vẽ sơ đồ mạng***

**Bước 3:** *Vẽ đỉnh 0 và các công việc đi ra từ nó.* Vẽ tất cả các công việc của WBS không đi sau một công việc nào đi ra từ đỉnh 0 này. Loại các công việc vừa được vẽ khỏi bảng WBS.

**Bước 4:** *Vẽ đỉnh thứ n, chụm các công việc đến nó và thêm các công việc đi ra từ nó.*

Giả sử ta đã vẽ được đến đỉnh thứ (n-1) (đỉnh n-1 đầu tiên là đỉnh 0) và đã vẽ tất cả các công việc đi ra từ nó.

Xét các công việc còn lại trong bảng WBS, có thể xảy ra ba trường hợp:

**Trường hợp 1:** *Thêm đỉnh kết thúc n:* Nếu các công việc của bảng WBS đã bị loại hết; *Thêm đỉnh thứ n*, và cho tất cả các công việc của sơ đồ chưa có đỉnh kết thúc chụm lại (kết thúc) tại **n**. **n** là đỉnh cuối cùng của mạng AOA. Quá trình vẽ mạng kết thúc.

**Trường hợp 2:** *Thêm đỉnh trung gian n:* Tìm **một bộ công việc** được đánh dấu ở cột “*Công việc đi trước*” của bảng WBS thỏa mãn hai điều kiện:

- Nằm trên cùng dòng với một công việc chưa bị loại (chưa vẽ)
- Chứa các công việc đã được vẽ trong sơ đồ.

**Thêm đỉnh trung gian thứ n** làm sự kiện kết thúc cho các công việc thuộc bộ này, tức là chụm chúng lại ở đỉnh **n**. *Tìm tất cả các công việc của WBS chưa được vẽ (chưa bị loại)*, mà trong cột “*Công việc đi trước*” trên cùng dòng với chúng có chứa bộ công việc vừa được xác định. *Vẽ những công việc này đi ra từ đỉnh n* và (*đánh dấu*) loại chúng khỏi bảng WBS.

**Trường hợp 3:** *Thêm đỉnh trung gian n giả:* Nếu bộ tìm được như trường hợp 2 có các công việc đều bị xóa, thì **thêm đỉnh trung gian n, nhưng là đỉnh giả**. Trong trường hợp này, chỉ vẽ các công việc đi ra từ nó như đã làm ở trường hợp 2 (không có công việc đi vào nó), và loại chúng ra khỏi bảng WBS.

Lặp lại **bước 4** cho đến khi các công việc của bảng WBS đã được vẽ (loại) hết.

#### **Bước 5:** Xác định các công việc giả

Các công việc giả chỉ liên quan đến các bộ công việc chứa công việc bị xóa ở cột “*Công việc đi trước*” của WBS.

Xét một công việc của WBS, mà trong bộ công việc cùng dòng với nó ở cột “*Công việc đi trước*” có một hay một bộ công việc bị xóa thì ta thêm một công việc giả từ đỉnh ngay sau công việc/bộ công việc bị xóa đó đến đỉnh mà công việc được xét này đi ra.

Chú ý: *Nếu bộ công việc* tương ứng với một công việc được xét có một số bộ công việc bị xóa, khi đó *sẽ có một số công việc giả* đi đến đỉnh mà công việc được xét này đi ra.

### **c. Giai đoạn 3: Đánh số các đỉnh mạng**

#### **Bước 6:** *Đánh số các đỉnh sơ đồ mạng theo yêu cầu*

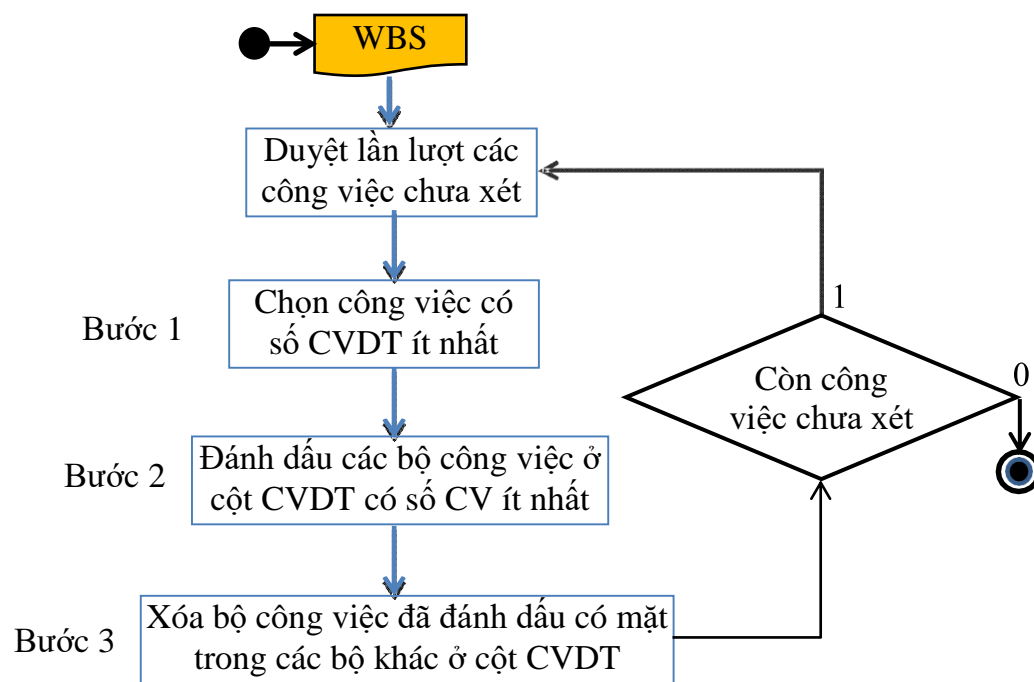
Đánh số 0 cho đỉnh khởi đầu và đánh số 1, 2, ... cho các đỉnh tiếp theo (đỉnh trung gian) sao cho đảm bảo nguyên tắc: *Số của đỉnh cuối công việc phải lớn hơn số đỉnh đầu công việc.*

Việc đánh số theo quy tắc trên là cơ sở cho việc tính toán các tham số thời gian của mạng sau này (bằng cách tính tham số thời gian cho các đỉnh theo thứ tự tăng hay giảm dần). Việc đánh số các đỉnh của mạng AOA là không duy nhất, trừ đỉnh đầu và đỉnh cuối cùng.

Như vậy, đến đây ta đã hoàn thành việc vẽ một sơ đồ một mạng dạng AOA. Mạng này sẽ được sử dụng cho việc lập kế hoạch lịch của dự án.

### **2.2.2. Sơ đồ khái niệm của thuật toán lập mạng bằng tay**

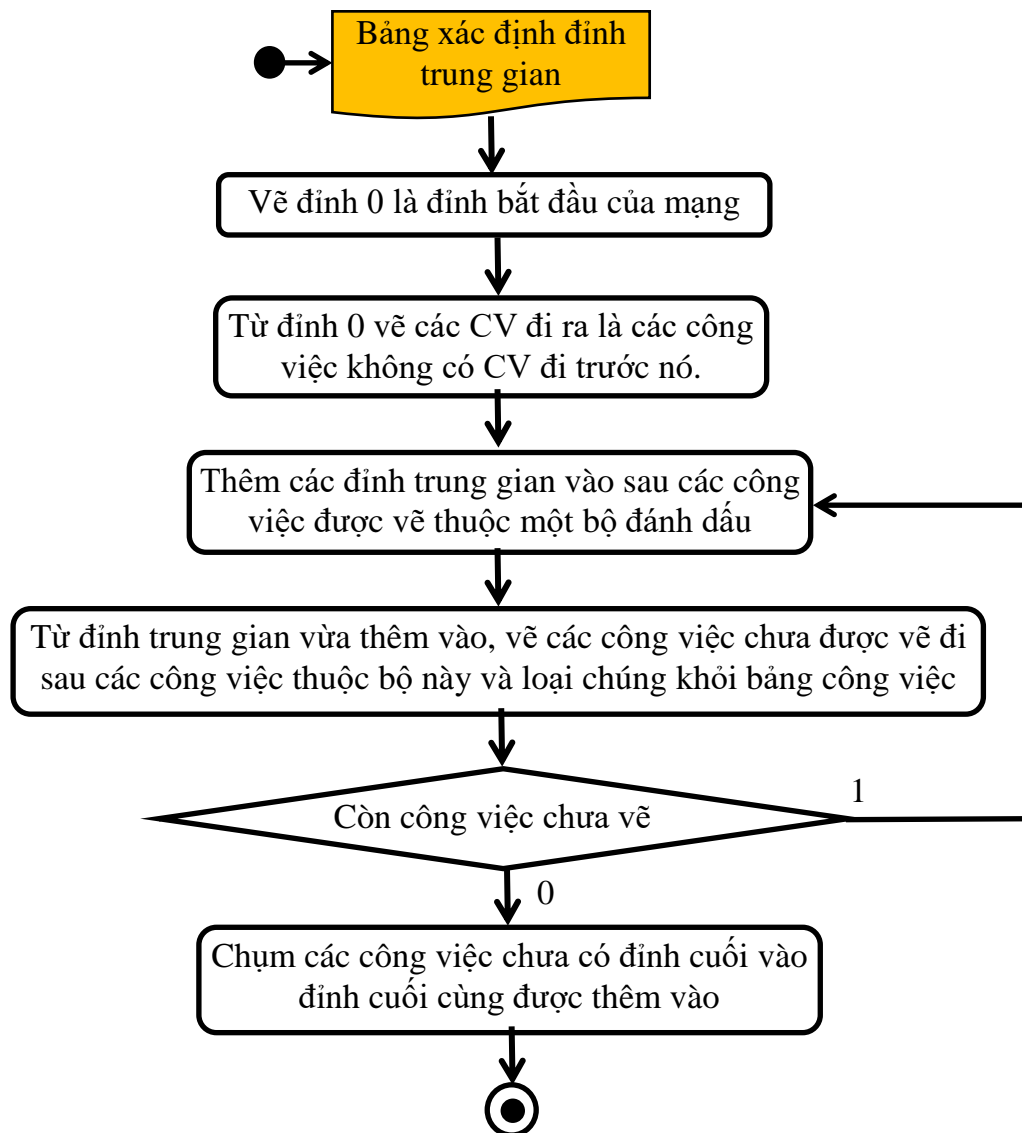
#### **a. Giai đoạn 1: xác định đỉnh trung gian**



Hình 2.1: Sơ đồ khái niệm xác định các đỉnh trung gian

***b. Giai đoạn 2: vẽ sơ đồ mạng***

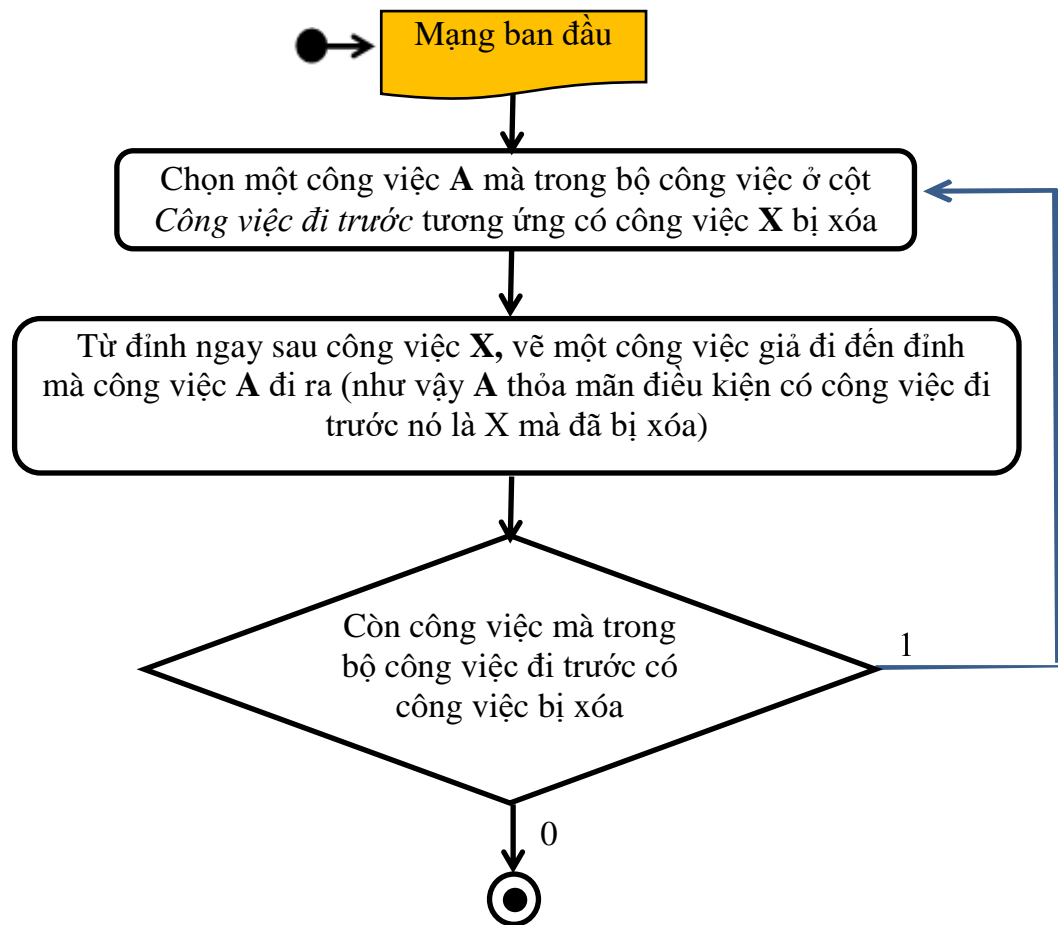
**b1. Vẽ sơ đồ mạng ban đầu**



Hình 2.2: Sơ đồ khái niệm vẽ mạng ban đầu

## b2. Thêm các công việc giả đảm bảo các ràng buộc của công việc

Mạng ban đầu được vẽ không thỏa mãn “*ràng buộc ban đầu*” về các công việc đi trước. Vì trong số các công việc đi trước (của một công việc đã cho) ở cột các “*Công việc đi trước*” đã bị xóa. Vì vậy, cần thêm các công việc giả để đảm bảo ràng buộc này: “mỗi công việc phải đi sau mọi công việc đi trước nó”.



Hình 2.3: Sơ đồ khái niệm thêm công việc giả vào mạng ban đầu

### 2.2.3. Ví dụ minh họa thuật toán lập mạng bằng tay

Giả sử cho một bảng phân rã công việc WBS có nội dung như ở bảng 2.1

Bảng 2.1: Bảng phân rã công việc

<i>Mã CV</i>	<i>Tên công việc</i>	<i>Công việc đi trước</i>	<i>Thời gian thực hiện</i>	<i>Nguồn lực sử dụng</i>
1	a	-	3	1
2	b	-	5	2
3	c	-	4	1
4	d	b	3	1
5	e	c	2	1
6	f	b, c	4	1
7	g	a	2	1
8	h	a, d	3	1
9	i	a, d	2	2
10	k	g	2	1
11	l	h, k	1	1

Kết quả quá trình vận dụng thuật toán đề xuất ở phần 2.1.1. tiến hành trên bảng 2.1 được diễn giải cụ thể như sau:

**a. Đánh dấu xác định các đỉnh trung gian**

(trong giai đoạn này chỉ xét các bộ công việc ở “Cột công việc đi trước”)

**Bước 1:** Trong cột “*Công việc đi trước*” của bảng công việc (bảng 2.2) có 4 dòng 4, 5, 7, 10 có các bộ công việc, mỗi bộ chỉ gồm **một** công việc là: **a, b, c** và **g**, nên ta đánh dấu chúng (bằng cách đưa vào trong ngoặc). Kết quả cho ở cột “*Bước 1*” của bảng 2.2.

**Bước 2:** Trong cột “*Công việc đi trước*” của bảng công việc, dòng 6 chứa hai bộ công việc đã được đánh dấu ở bước 1 là (b) và (c); dòng 8 và 9 chứa bộ (a) đã đánh dấu ở bước 1, ta xóa đi những bộ này (xem kết quả ở cột “*Bước 2*” bảng 2). Loại các bộ công việc đi trước đã đánh dấu hoặc bị xóa, chuyển sang bước 1’.

**Bước 1’:** Trong cột “*Công việc đi trước*” của bảng bây giờ chỉ còn 3 dòng (8, 9, 11): trong đó ở dòng 8 và 9 có bộ công việc chỉ gồm một công việc là **d**, ta đánh dấu hai bộ một công việc này (xem cột “*Bước 1’*” bảng 2.2).

**Bước 2’:** Trong bảng công việc sau đánh dấu, không có dòng nào chứa bộ công việc (d) vừa đánh dấu ở bước 1’, nên không cần thực hiện thao tác xóa (xem cột “Bước 2” trong bảng 2.2). Chuyển sang bước 1”.

**Bước 1”:** Trong cột “*Công việc đi trước*” của bảng công việc bây giờ chỉ còn duy nhất một dòng 11 với *bộ có 2 công việc {h,k}*, ta đánh dấu chúng. Giai đoạn 1 kết thúc vì trong cột “*Công việc đi trước*” không còn dòng nào (xem cột “Bước 2” trong bảng 2.2)

Như vậy, kết quả việc đánh dấu các bộ công việc và xóa chúng trong giai đoạn 1 được cho ở bảng 2.3. Trong đó có 6 bộ công việc: (a), (b), (c), (d), (g) và (h,k) được đánh dấu (bằng cách đưa vào trong ngoặc đơn) và những bộ công việc khác bị loại (bằng cách dùng đoạn thẳng để xóa chúng). Các bộ công việc được đánh dấu xác định 6 đỉnh trung gian của mạng AOA tương ứng với bảng 2.1

Bảng 2.2: Thực hiện các bước của giai đoạn 1

mã CV	Tên công việc	<i>Bước1</i>	<i>Bước2</i>	<i>Bước1’</i>	<i>Bước2’</i>	<i>Bước1”</i>	<i>Bước2”</i>
		Công việc đi trước	Công việc đi trước	Công việc đi trước	Công việc đi trước	Công việc đi trước	Công việc đi trước
4	d	(b)	(b)				
5	e	(c)	(c)				
6	f	b,c	<del>(b),(c)</del>				
7	g	(a)	(a)				
8	h	a,d	<del>(a), d</del>	<del>(a),(d)</del>			
9	i	a,d	<del>(a), d</del>	<del>(a),(d)</del>			
10	k	(g)	(g)				
11	l	h, k	h, k	h, k	h, k	(h,k)	



Bảng 2.3: Kết quả thực hiện các bước của giai đoạn 1

Mã CV	Tên công việc	Công việc đi trước	Mã CV	Tên công việc	Công việc đi trước
1	a	-	7	g	(a)
2	b	-	8	h	<del>(a)</del> , (d)
3	c	-	9	i	<del>(a)</del> , (d)
4	d	(b)	10	k	(g)
5	e	(c)	11	l	(h,k)
6	f	<del>(b)</del> , <del>(c)</del>			

**b. Vẽ sơ đồ mạng công việc**

**Bước 3:** Vẽ đỉnh đầu tiên đánh số 0. Từ đỉnh 0, vẽ 3 công việc đi ra là **a**, **b**, **c**: là những công việc không đi sau một công việc nào. Loại các công việc a, b, c đã được vẽ ra khỏi bảng.

**Bước 4a:** Vì công việc **a** đã được vẽ và bộ (a) được đánh dấu, nên ta thêm đỉnh **1** sau **a** (là kết thúc của **a**), và có duy nhất công việc **g** đi sau **a**, ta vẽ **g** từ đỉnh **1**. Loại **g** khỏi bảng.

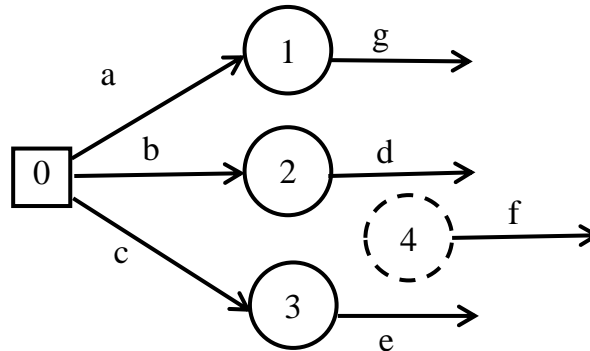
**Bước 4b:** Vì công việc **b** đã được vẽ và bộ (b) được đánh dấu, nên ta thêm đỉnh **2** sau **b** (là kết thúc của **b**), và có duy nhất công việc **d** đi sau **b**, ta vẽ **d** từ đỉnh **2**. Loại **d** khỏi bảng.

**Bước 4c:** Vì công việc **c** đã được vẽ và bộ (c) được đánh dấu, nên ta thêm đỉnh **3** sau **c**. Có duy nhất công việc **e** đi sau **c**, ta vẽ **e** từ đỉnh **3**. Loại **e** khỏi bảng.

Bảng 2.4: Bảng công việc còn lại sau 4 lần lặp lại bước 4 của giai đoạn 2.

TT	Tên công việc	Công việc đi trước
8	h	<del>(a)</del> , (d)
9	i	<del>(a)</del> , (d)
10	k	(g)
11	l	(h,k)

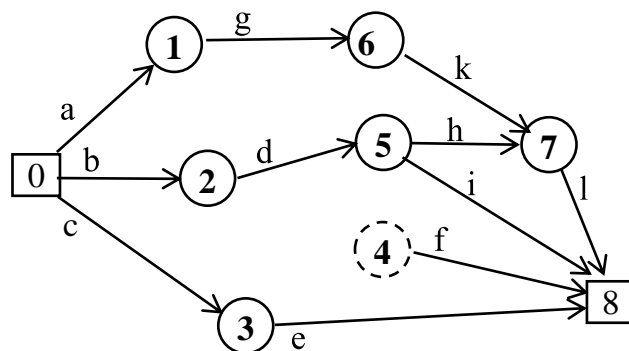
**Bước 4d:** Bộ công việc (b,c) gồm hai công việc b và c đã được vẽ, cùng dòng với công việc f chưa được vẽ, nhưng cả b và c này đều đã bị xóa. Vậy cần thêm một *đỉnh giả* 4, và chỉ có f đi sau (b,c), ta vẽ f đi ra từ đỉnh giả này. Loại f khỏi bảng. Kết quả nhận được đến bước này cho ở bảng 2.4 và hình 2.1.



Hình 2.4: Mạng công việc AOA sau khi kết thúc bước 4d

Tiếp tục, vì các bộ (d) và (g) được đánh dấu và d, g đã được vẽ, trên các dòng tương ứng với công việc d và g chưa vẽ, nên ta thêm đỉnh 5 sau d và đỉnh 6 sau g. Vì có hai công việc h và i đi sau d, ta vẽ hai công việc này đi ra từ đỉnh 5. Tương tự, vì công việc k đi sau g, ta vẽ k đi ra từ đỉnh 6.

Công việc l chưa vẽ, cùng dòng với bộ {h, k} mà công việc h và k đã được vẽ, nên ta vẽ đỉnh 7 là kết thúc của hai công việc h và k. Đến đây tất cả các công việc đã được vẽ, ta thêm đỉnh 8 vào cho các công việc chưa có đỉnh kết thúc là e, f, i, l kết thúc (chụm lại) tại đây (xem hình 2.2).



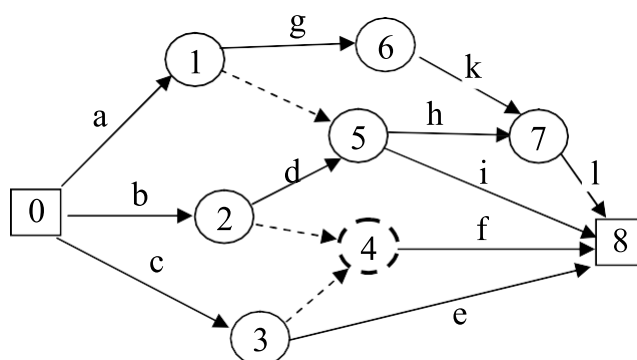
Hình 2.5: Mạng công việc AOA sau khi kết thúc bước 4

**Bước 5:** Dòng 8 ở cột “*Công việc đi trước*” có bộ (d) xác định đỉnh 5 và bộ (a) bị xóa, nên cần thêm công việc giả từ đỉnh 1 sau (a) đến đỉnh 5. Cũng tương tự, ở dòng 6 có bộ (b) và (c) tất cả bị xóa, đã thêm đỉnh giả 4 (ở bước 4), nên cần thêm công việc giả tương ứng với (b) từ đỉnh 2 (sau b) đến đỉnh 4 và công việc giả tương ứng với (c) từ đỉnh 3 (sau c) đến đỉnh giả 4. Bước 5 kết thúc, vì cột “*Công việc đi trước*” của bảng không còn dòng nào có bộ công việc bị xóa. Mạng công việc AOA kết quả cho ở hình 2.3. Mạng có 3 công việc giả và một đỉnh giả.

### c. Đánh số các đỉnh của mạng

**Bước 6:** Các đỉnh của mạng đã thỏa mãn yêu cầu đánh số đặt ra (đỉnh cuối công việc phải lớn hơn đỉnh đầu công việc) nên không cần đánh số lại.

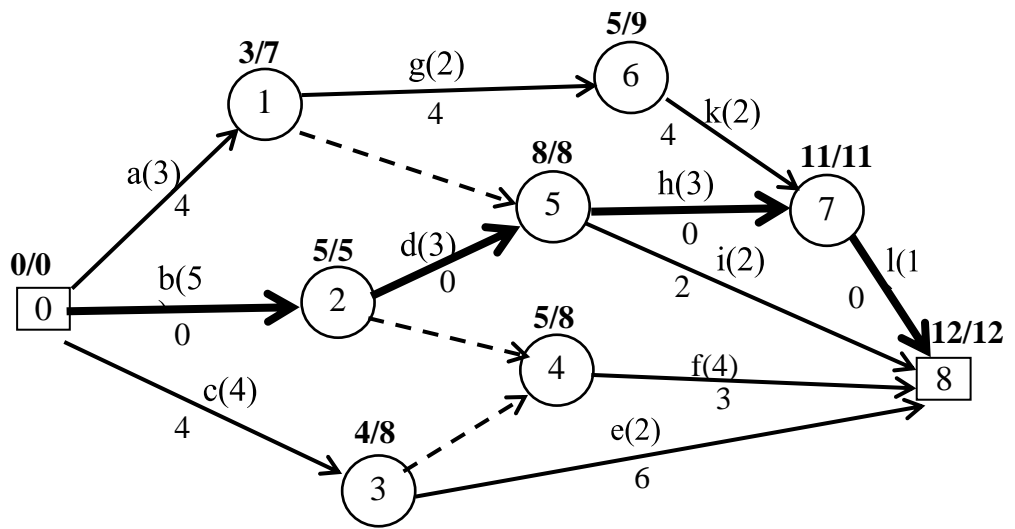
Kết thúc thuật toán, ta vẽ được mạng cho ở hình 2.6.



Hình 2.6: Mạng công việc AOA sau khi kết thúc bước 6

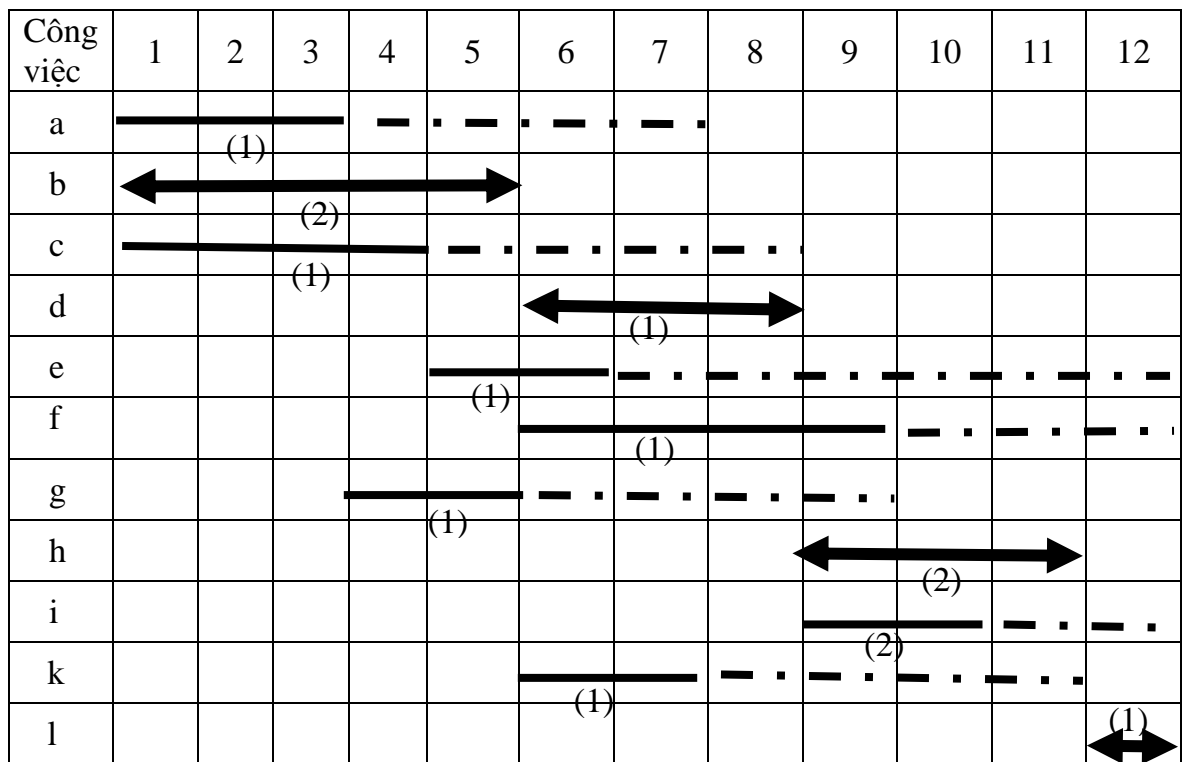
### 2.2.4. Sử dụng mạng lập được để lập lịch dự án

Mạng trên hình 2.6 được sử dụng để tính toán các tham số thời gian trực tiếp trên nó (xem trên hình 2.7). Các tham số thời gian của mạng là cơ sở để lập kế hoạch lịch. Trong kế hoạch lịch, sơ đồ Gantt (xem hình 2.8) và sơ đồ sử dụng nguồn lực (2.9) được xây dựng từ mạng các tham số thời gian (hình 2.7). Chúng là các công cụ trợ giúp một cách hiệu quả cho việc đánh giá và quản lý điều hành dự án.

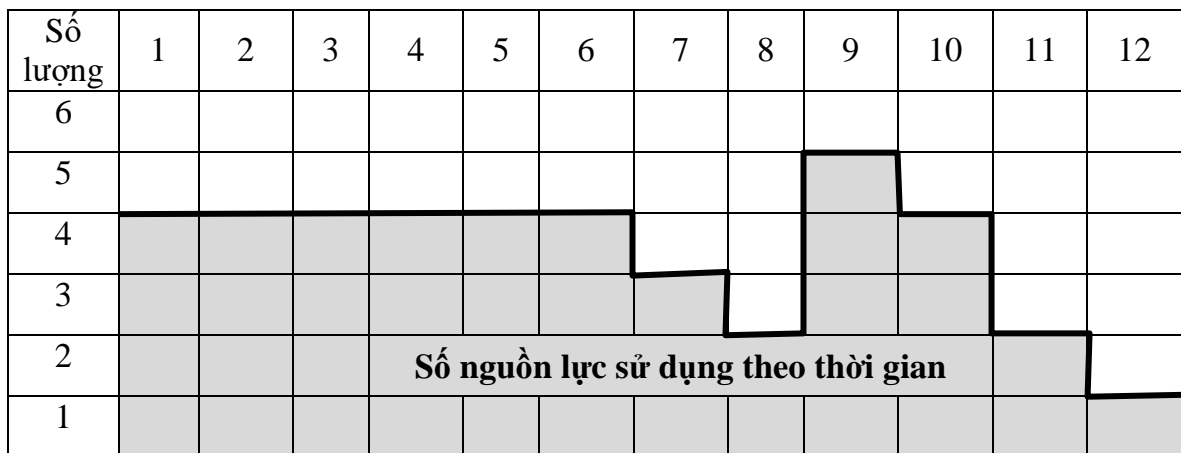


**Ghi chú:** Số ghi bên tên cạnh công việc là thời gian thực hiện, dưới công việc là thời gian dự phòng của nó. Số ghi trên các đỉnh mạng là: thời gian bắt đầu sớm nhất ( $t_s$ )/thời gian kết thúc muộn nhất ( $t_m$ ) của đỉnh đó. Công việc biểu diễn bằng mũi tên đậm là công việc Gantt. Công việc với mũi tên nét đứt là công việc giả có thời gian thực hiện bằng 0.

Hình 2.7: Mạng công việc với các tham số thời gian được tính toán



Hình 2.8: Biểu đồ Gantt kế hoạch lịch của ví dụ



Hình 2.9. Biểu đồ sử dụng nguồn lực (người) của ví dụ.

## **Chương III: THIẾT KẾ THUẬT TOÁN CHO VIỆC TỰ ĐỘNG TÍNH TOÁN DỰ ÁN**

### **3.1. Tính toán đánh giá khả thi kinh tế của dự án**

Do việc tính toán hệ số đánh giá khả thi kinh tế của dự án là tương đối đơn giản, nên có thể tổ chức nó trên các bảng Excel để tiện cho người dùng (vì nhiều người biết dùng Excel) và dễ cài đặt bảng tính.

#### **3.1.1. Các tham số để tính toán hệ số hoàn vốn**

Các tham số cho việc tính hệ số hoàn vốn bao gồm **5** đại lượng:

1. *Tổng vốn đầu tư ban đầu (TV)*: thường được tổng hợp từ giá trị của các yếu tố thành phần như nhà cửa, vật kiến trúc, trang thiết bị, ... tham gia vào thực hiện dự án.
2. *Số  $n$ , là số chu kỳ* được chọn mà dự án cần phải thực hiện. Nó được lựa chọn đủ dài để có thể hoàn vốn, nhưng cũng không vượt quá thời gian có thể sử dụng của những tài sản cố định dùng cho dự án (vì theo giả thiết chúng chỉ đầu tư một lần).
3. *Tỷ lệ lãi suất vốn vay  $r$*  (thường gọi là *tỷ lệ chiết khấu* tính theo %), thường lớn hơn lãi suất tiết kiệm. Khi đầu tư ta luôn xem nguồn vốn là đi vay, vì thế cần sử dụng lãi suất vay để tính toán.
4. *Chi phí thường xuyên* cho mỗi kỳ để tạo ra sản phẩm của dự án ( $CF_i$ )
5. *Doanh thu thường xuyên* mỗi kỳ do tiêu thụ sản phẩm của dự án ( $DT_i$ )

#### **3.1.2. Cấu trúc bảng tính toán phân tích khả thi kinh tế**

Bảng tính toán được thiết kế cho ở bảng 3.1.

Bảng 3.1. Bảng tính toán hệ số hoàn vốn (phân tích chi phí - hiệu quả)

Khoản mục	Chu kỳ 0	Chu kỳ 1	...	Chu kỳ n	Tổng cộng
Số chu kỳ tính	<b>n</b>				
Hệ số chiết khấu	<b>R%</b>				
1. Hệ số quy đổi	1	$1/(1+r)$	...	$1/(1+r)^n$	
2. Thu nhập trong kỳ		<b>DT<sub>1</sub></b>	...	<b>DT<sub>n</sub></b>	
3. Thu nhập kỳ quy đổi (1x2)		$DT_1/(1+r)$	...	$DT_n/(1+r)^n$	$\Sigma DT_{qđ}$
4. Thu nhập quy đổi tích lũy = $(4_{.1} + 3)$	0	DTTL <sub>1</sub>	...	DTTL <sub>n</sub>	
<b>Các khoản chi</b>					
5. Chi ban đầu (1 lần- đầu tư)	<b>TV</b>				
6. Chi trong kỳ		<b>CF<sub>1</sub></b>	...	<b>CF<sub>n</sub></b>	
7. Chi trong kỳ quy đổi (6x2)		$CF_1/(1+r)$	...	$CF_n/(1+r)^n$	$\Sigma CF_{qđ}$
8. Chi quy đổi tích lũy = $(8_{.1} + 7)$	0	CFTL <sub>1</sub>	...	CFTL <sub>n</sub>	
<b>Hiệu quả</b>					
9. Lãi ròng quy đổi kỳ = (3-7)		Lai <sub>1</sub>	...	Lai <sub>n</sub>	$\Sigma DT_{qđ} - \Sigma CF_{qđ}$
10. Hệ số hoàn vốn					ROI
11. Thời gian hoàn vốn					T= 1/ROI

**Công thức tính:  $ROI = (\text{lợi nhuận bình quân trong kỳ}) / \text{tổng vốn đầu tư} = (\Sigma DT_{qđ} - \Sigma CF_{qđ}) / n / TV$**

- Bảng được thiết kế sẵn các công thức tính trong các ô, ta chỉ cần nhập dữ liệu vào **3** ô màu sẫm ở cột kỳ 0 và các ô ở **2** dòng: dòng **2** (*thu nhập*) và dòng **6** (*chi phí*) là ta có ngay kết quả tính toán ở các ô còn lại.

- Sử dụng 2 dòng **4** và **8** ta có thể tính được thời gian hoàn vốn bằng phương pháp đồ thị, từ đó suy ra **ROI**.

### 3.1.3. Ví dụ tính toán phân tích khả thi kinh tế

Giả sử, cần đầu tư kinh doanh một quán cà phê. *Đầu tư ban đầu*: (thuê địa điểm (6tháng): 30triệu) + (mua sắm trang thiết bị: 14 triệu) = **44** triệu. *Chi thường xuyên*: (nguyên vật liệu: 9 triệu/tháng) + (điện, nước: 2,5 triệu/tháng) + (nhân công:4,5 triệu/tháng) = **16** triệu/tháng. *Doanh thu tháng* (1triệu/ngày)x(30 ngày) = **30** triệu/tháng. Tính trong **6** tháng, **r = 9%**/tháng.

BẢNG PHÂN TÍCH KINH TẾ - TÍNH HỆ SỐ HOÀN VỐN								
Đơn vị tính: triệu đồng								
Khoản mục	Tháng 0	Tháng 1	Tháng 2	Tháng 3	Tháng 4	Tháng 5	Tháng 6	Tổng số
Số chu kỳ tính	<b>6 tháng</b>							
Hệ số chiết khấu	<b>9% tháng</b>							
1. Hệ số quy đổi	1	0.9174	0.8417	0.7722	0.7084	0.6499	0.5963	
2. Thu hàng tháng	0	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>	
3. Thu quy đổi tháng = (1x2)		27.5229	25.2504	23.1655	21.2528	19.4979	17.8880	134.5776
4. Thu quy đổi tích lũy = (4(-1)+3)	0	27.5229	52.7733	75.9388	97.1916	116.6895	134.5776	
Các khoản chi								
5. Chi ban đầu	<b>44</b>							
6. Chi hàng tháng		<b>16.0</b>	<b>16.0</b>	<b>16.0</b>	<b>16.0</b>	<b>16.0</b>	<b>16.0</b>	
7. Chi quy đổi tháng (6x2)		14.6789	13.4669	12.3549	11.3348	10.3989	9.5403	71.7747
8. Chi quy đổi tích lũy (8 (-1)+7)	44	58.6789	72.1458	84.5007	95.8355	106.2344	115.7747	
9. Lãi ròng tháng (3-7)		12.8440	11.7835	10.8106	9.9180	9.0990	8.3477	62.8029
10. Hệ số hoàn vốn ( <i>lợi nhuận bq</i> )	<b>10.4671</b>							<b>0.24</b>
11. Thời gian hoàn vốn (tháng)								<b>4.20</b>

Hệ số hoàn vốn tính được là **0,24**, tức là cứ một đồng vốn đầu tư trong 6 tháng cho trung bình 0,24 đồng lãi. Và thời gian hoàn vốn là **4,2** tháng. Tức là, chỉ sau 4 tháng 6 ngày, ta thu hồi được toàn bộ vốn bỏ ra ban đầu, và sau đó lãi làm ra ta được hưởng cả.

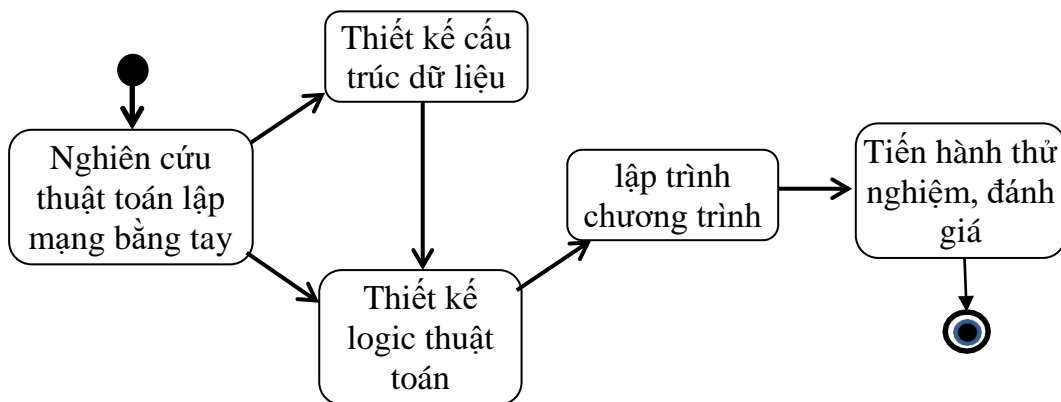


## 3.2. Chuyển thuật toán lập kế hoạch dự án làm tay sang làm máy

Thuật toán lập mạng ở chương trước được xây dựng để làm bằng tay (làm thủ công). Sau khi lập mạng bằng tay, ta phải cập nhật dữ liệu mạng vào máy tính để tiến hành tính toán bản kế hoạch lịch cho quản lý dự án. Việc lập mạng bằng tay tốn rất nhiều thời gian, đặc biệt khi mạng có kích cỡ lớn. Hơn nữa, mỗi khi có thay đổi trong cấu trúc của mạng (như thêm công việc mới hay thay đổi trình tự giữa một số công việc) thì việc vẽ lại mạng phải làm lại từ đầu. Vì thế, khó có thể sử dụng công cụ vẽ mạng dạng AOA bằng tay cho hoạt động quản lý dự án, vì không đáp ứng yêu cầu về thời gian. Chương này trình bày quá trình chuyển thuật toán lập mạng bằng tay sang lập mạng trên máy để có thể tự động hóa việc lập kế hoạch lịch cho dự án

### 3.2.1. Sơ đồ tổng quát chuyển đổi thuật toán sang làm máy

Quá trình chuyển từ lập mạng bằng tay sang làm máy được mô tả ở hình 3.1.



Hình 3.1: Sơ đồ tiến trình chuyển sang lập mạng trên máy

### 3.2.2. Bảng cấu trúc dữ liệu cho thuật toán lập kế hoạch dự án trên máy

Bảng cấu trúc dữ liệu cho việc thực hiện thuật toán lập mạng trên máy và tính toán các tham số kế hoạch lịch được biểu diễn ở bảng 3.1. Nó gồm 9 cột và các dòng. Mỗi một dòng (i) tương ứng với một công việc.

Bảng 3.2. Bảng cấu trúc dữ liệu cho bài toán lập kế hoạch lịch trên máy

Mã CV	Tên công việc	CV đi trước	CV chọn	CV loại	Đỉnh đầu CV	Đỉnh cuối CV	Thời gian	Nhân lực
TT(i)	CV(i)	CVtr(i)	CVch(i)	CVlo(i)	ddCV(i)	dcCV(i)	tgCV(i)	nlCV(i)
1	a	-	-		0	1	3	1
2	b	-	-		0	2	5	2
3	c	a	a		1	4	4	1
4	d	a, b		a,b	3	4	3	1
5	e	b	b		2	4	2	1
6	f	b	b		2	6	4	1
7	g	e, d	e,d		4	7	2	1
8	h	e, d	e,d		4	6	3	1
9	i	c	c		7	7	2	2
10	k	c, h, f	h,f	c	6	7	2	1

- *Cột 1: Mã của công việc (TT(i) là mã công việc thứ i : có thể dùng là số nguyên i).*
- *Cột 2: Tên các công việc (tên công việc thứ i: CV(i)).*
- *Cột 3: Cột các công việc đi trước: gồm các công việc đi trước công việc (CV(i)) ở dòng này. Nó được biểu diễn bằng véc tơ CVtr(i). Mỗi thành phần của véc tơ này là một mã (hay tên) của một công việc đi trước công CV(i). Các thao tác chính của thuật toán sẽ thao tác trên các phần tử của cột này.*
- *Cột 4: Cột bộ công việc được chọn (hay đánh dấu): Khi thực hiện bước 2 của giai đoạn 1: xác định các đỉnh trung gian, bộ công việc CVch(i) được chọn lấy từ phần tử cùng dòng (CVtr(i)) ở trước cột “Công việc đi trước”, và loại chúng khỏi dòng của cột này.*
- *Cột 5 : Cột các công việc bị loại (CVlo), để lưu lại các công việc bị xóa CVlo(i) khỏi bộ công việc đi trước CVtr(i) của các công việc còn lại khi thực hiện bước 3 của giai đoạn 1: xác định các đỉnh trung gian.*

Như vậy, trong quá trình xác định đỉnh trung gian, bộ các công việc đi trước ở cột “Công việc đi trước” của một công việc i sẽ được chọn để “đánh dấu” hoặc “xóa hết”, điều đó có nghĩa là  $CVtr(i) = \emptyset$ . Vì thế, cột CVtr sẽ được dùng để kiểm

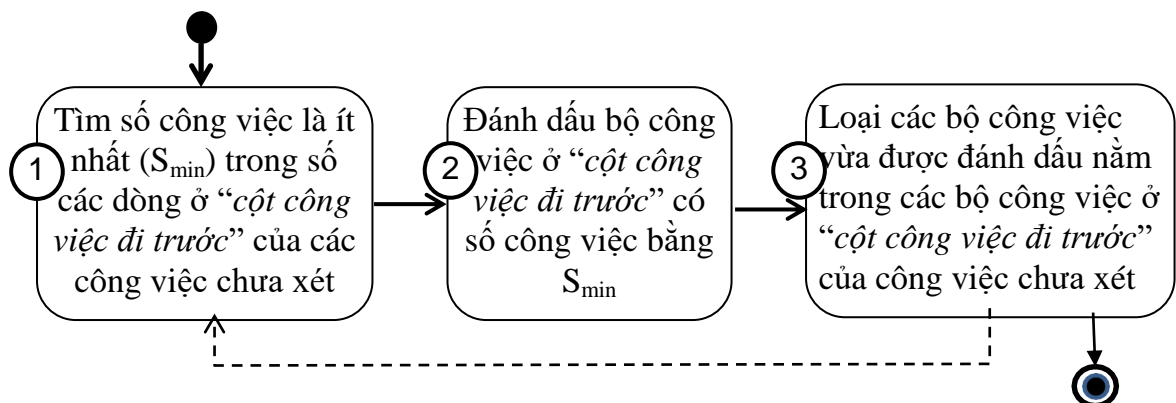
tra một công việc đã được xem xét ( $CV_{tr}(i) = \emptyset$ ) hay chưa ( $CV_{tr}(i) \neq \emptyset$ ). Bước xác định các đỉnh trung gian sẽ kết thúc khi mọi bộ công việc ở cột này đã xem xét (hay xóa đi), tức là  $CV_{tr}(i) = \emptyset$  với mọi  $i$ .

- Cột 6: *Cột đỉnh đầu của công việc* (ddCV). Véc tơ ddCV(i) lưu đỉnh bắt đầu của công việc  $i$ .
- Cột 7: *Cột đỉnh cuối của công việc* (dcCV). Véc tơ dcCV(i) lưu đỉnh cuối của mỗi công việc  $i$ .
- Cột 8: *Cột thời gian* (tgCV), là cột dữ liệu về thời gian thực hiện của công việc  $i$ : Cột này sẽ được sử dụng để tính các tham số về thời gian của mạng.
- Cột 9: Là cột *dữ liệu về nguồn lực* (nlCV) cần để thực hiện công việc  $i$ : Cột dữ liệu này sẽ được sử dụng để tính toán ở bước lập sơ đồ nguồn lực và cân đối nguồn lực.
- Cột 10: *Cột đánh dấu X* để đánh dấu công việc đã được xét khi thực hiện thuật toán

### 3.2.3. Thiết kế thuật toán cho chương trình lập mạng AOA

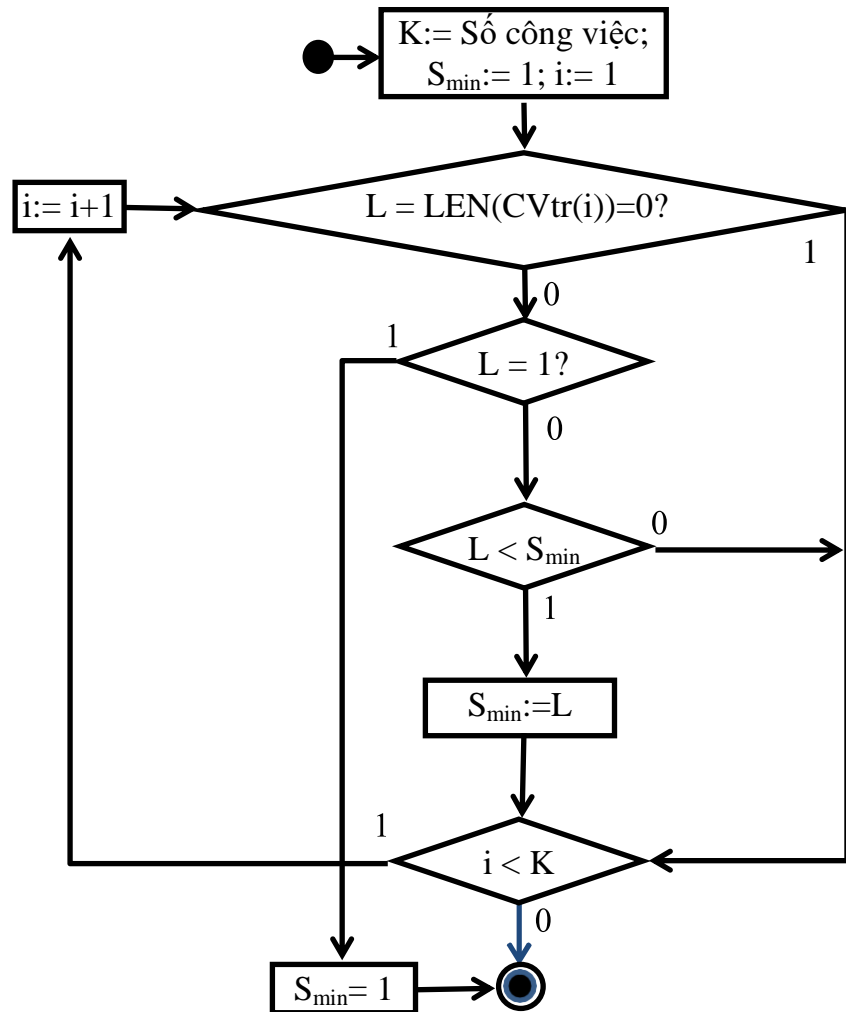
#### a. Sơ đồ thuật toán xác định đỉnh trung gian

Sơ đồ thuật toán xác định các đỉnh trung gian gồm 3 bước như ở hình 3.2.



Hình 3.2: Sơ đồ các bước xác định đỉnh trung gian

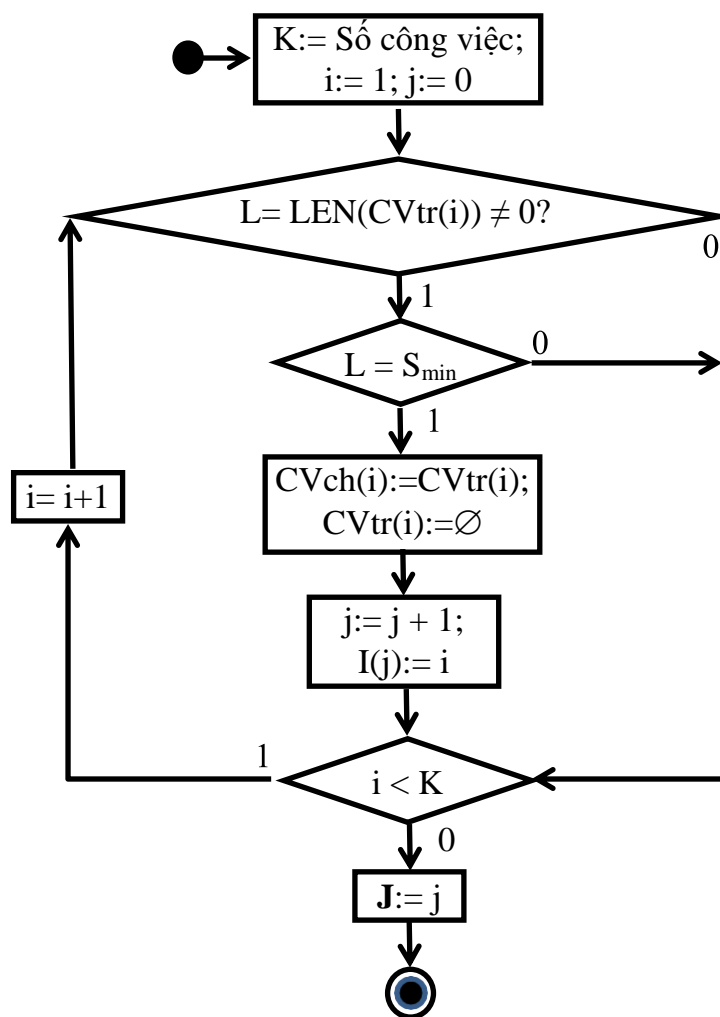
**Bước 1:** Tìm số công việc nhỏ nhất  $S_{\min}$  trong số các bộ công việc (ở một dòng) của cột “*Công việc đi trước*” mà chưa được xét (chưa chọn) (hình 3.3).



Hình 3.3: Tìm số công việc là nhỏ nhất của các dòng chưa xét

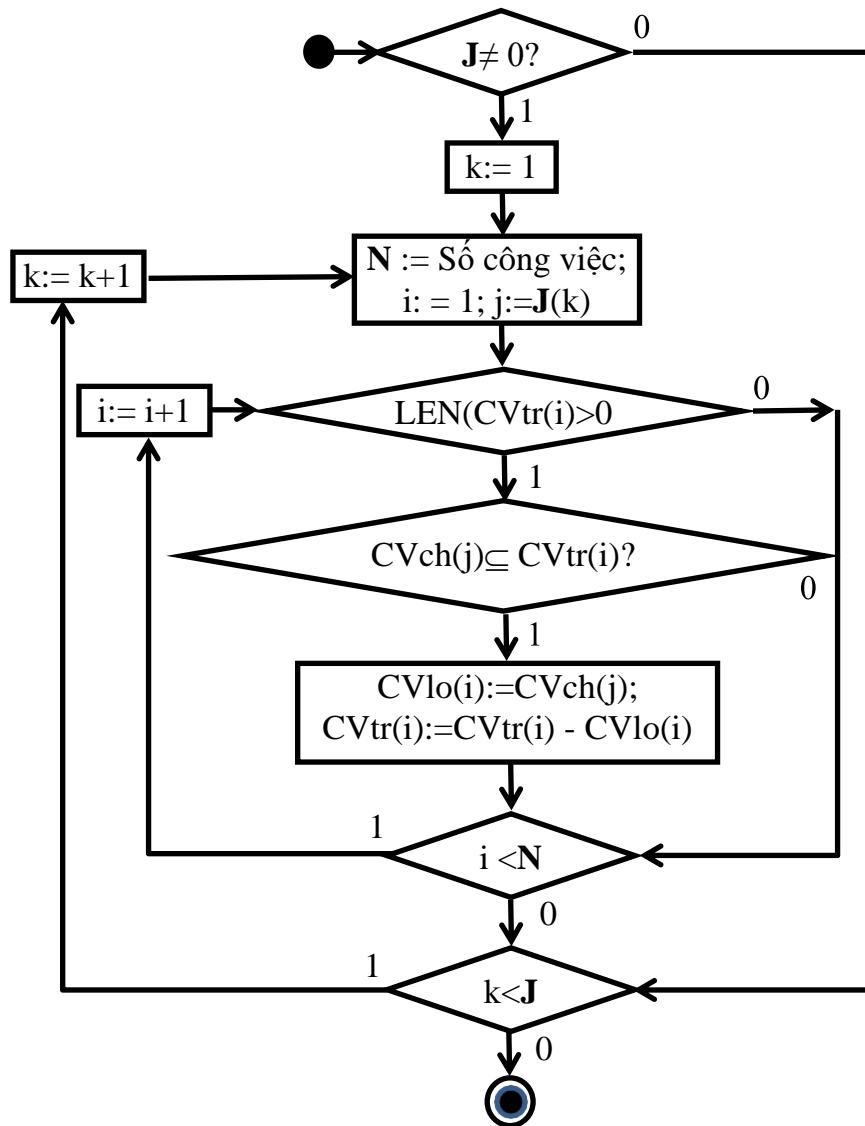
**Bước 2:** Đánh dấu các bộ công việc ở cột “*Công việc đi trước*” của công việc chưa xét và có số công việc là nhỏ nhất ( $= S_{\min}$ ) và chuyển chúng sang ở cột “*CVch*” (hình 3.4).

Ở đây, **J** là số bộ công việc đã được đánh dấu và xóa khỏi cột “*Công việc đi trước*” (CVtr), và chuyển sang cột “*Bộ công việc được chọn*” (CVch) với **I(i)** chứa chỉ số các dòng tương ứng với chúng



Hình 3.4: Đánh dấu các bộ công việc có CVDT là nhỏ nhất

**Bước 3:** Xóa các bộ công việc đã đánh dấu có mặt trong các bộ khác còn lại ở cột “*Công việc đi trước*” (hình 3.5).



Hình 3.5: Xóa bộ công việc đã đánh dấu có mặt trong các bộ khác

**b. Sơ đồ thuật toán vẽ sơ đồ mạng**

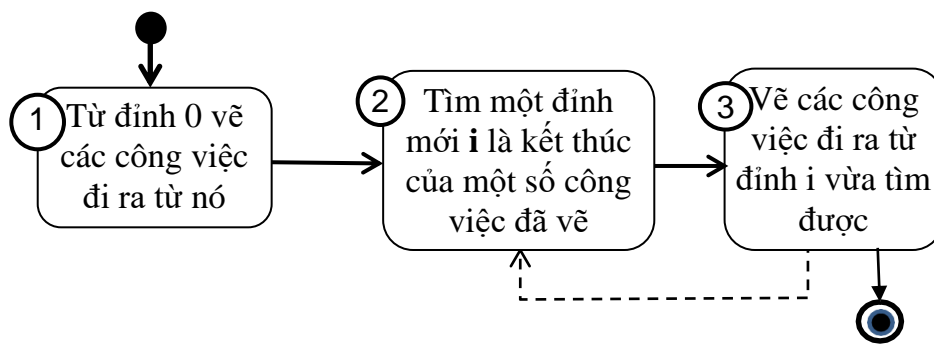
Để thực hiện việc vẽ sơ đồ mạng chúng ta cần phải sử dụng các dữ liệu:

- Cột đánh dấu các công việc đã được vẽ: cột 10 bảng 2.5. Trong đó, công việc  $i$  chưa vẽ có  $X(i)=1$ , đã vẽ:  $X(i)=0$ .

- Dữ liệu ở cột “*Cột bộ công việc chọn*” (CVch – đã được xác định ở 3.3.1) là cơ sở để xác định đỉnh trung gian và “*Cột công việc loại*” (CVlo) là cơ sở để xác định các đỉnh giả.

Trong giai đoạn này, ta cần xác định đỉnh bắt đầu của tất cả các công việc  $i$  (ddCV( $i$ )) cũng như các đỉnh kết thúc của nó (dcCV( $i$ )). Quá trình được tiến hành dần theo vết dầu loang: từ đỉnh 0 ta vẽ các công việc đi ra khỏi nó. Sau đó tìm một đỉnh trung gian mới là kết thúc cho một số công việc đã vẽ. Và từ đỉnh mới tìm được ta tiếp tục quá trình vẽ các công việc mới đi ra từ nó... cho đến khi vẽ hết tất cả các công việc.

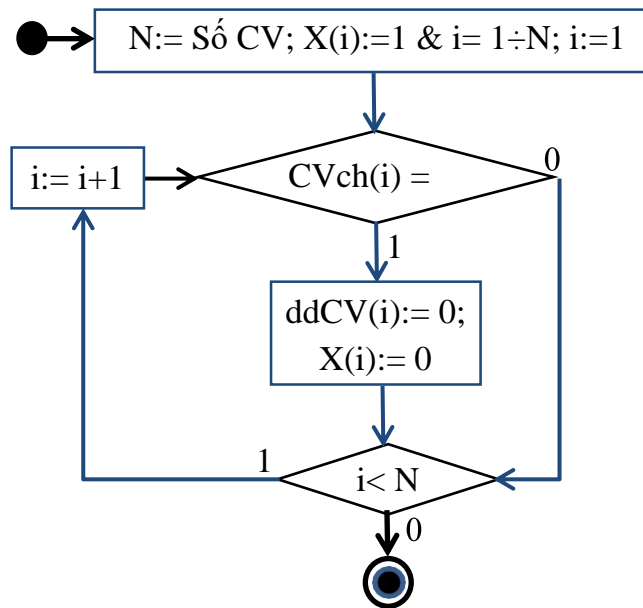
Sơ đồ quá trình vẽ mạng được mô tả ở hình 3.6.



Hình 3.6. Sơ đồ vẽ mạng: xác định dần các đỉnh đầu, cuối của các công việc

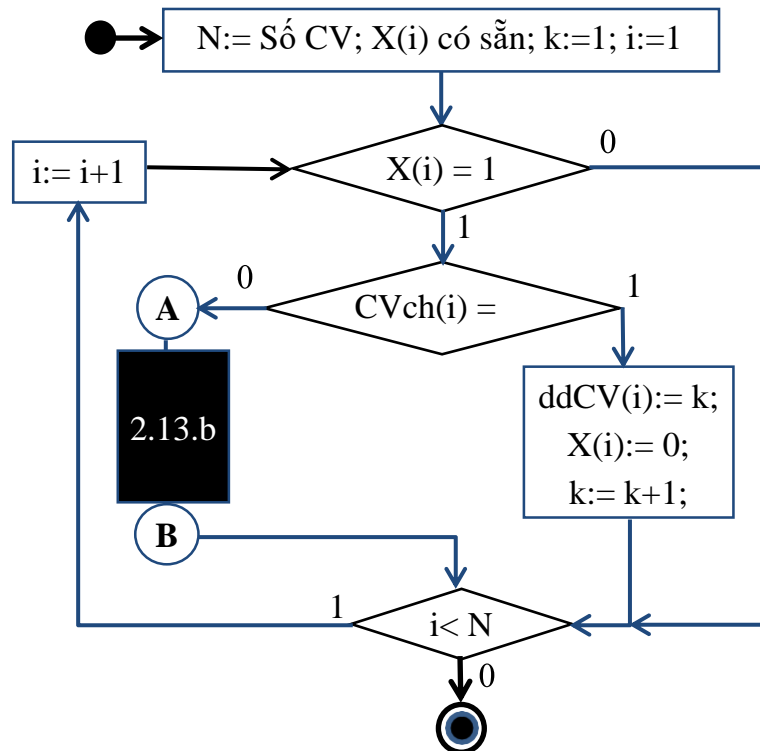
Sơ đồ logic thuật toán để vẽ sơ đồ mạng bao gồm:

**Bước 1:** Từ đỉnh 0, xác định các công việc đi ra từ nó là các công việc không có công việc đi trước (hình 3.7).



Hình 3.7: Thêm đỉnh 0 và vẽ các công việc đi ra từ nó

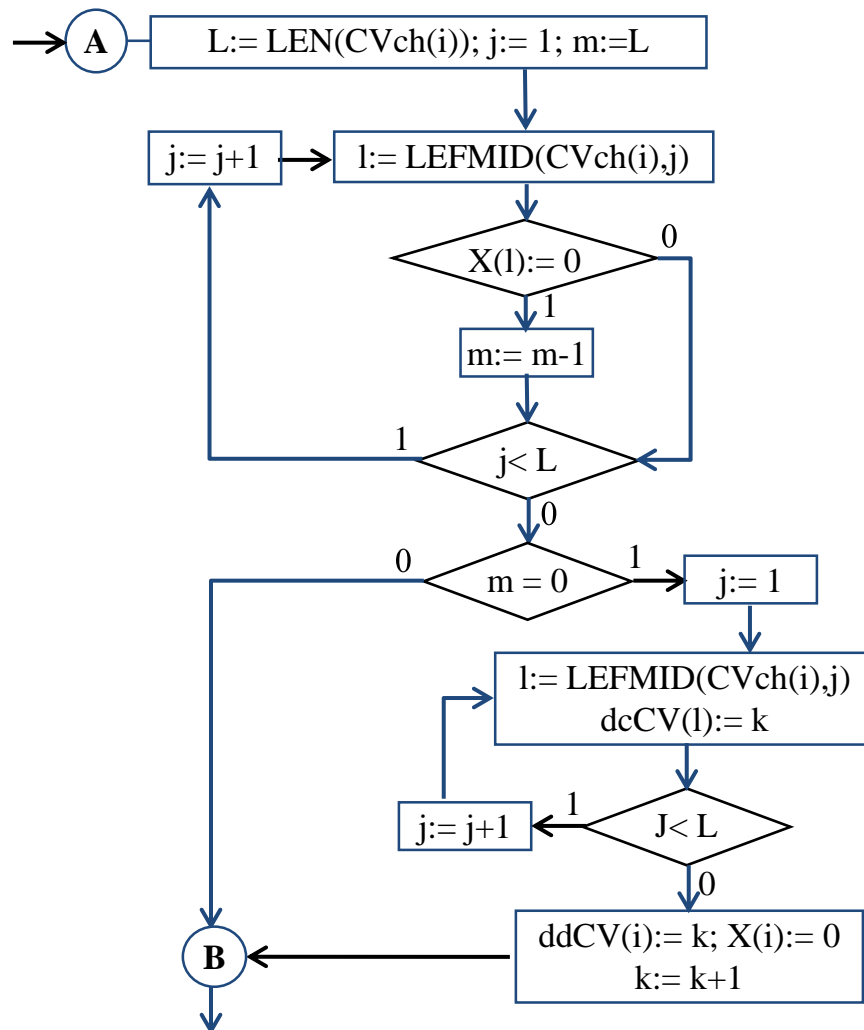
**Bước 2:** Thêm đỉnh trung gian cho một số công việc đã vẽ, đánh số đỉnh cuối cho các công việc (hay thêm đỉnh giả) và vẽ công việc đi ra từ đỉnh này (hình 3.8a).



Hình 3.8a: Thêm đỉnh trung gian **k** và vẽ công việc đi ra từ **k**

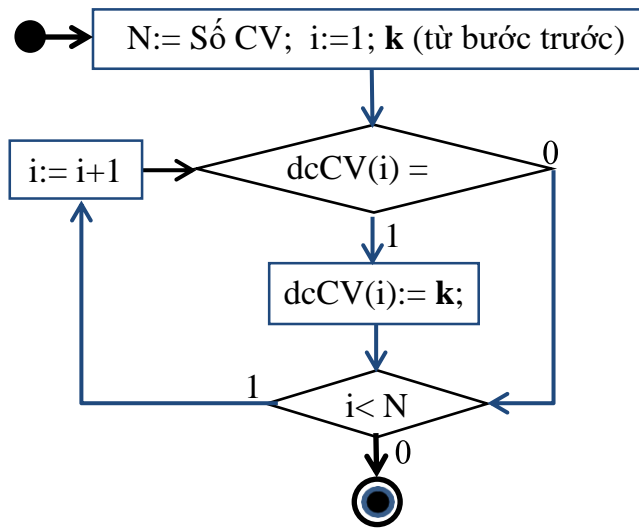


Trong hình 3.8b,  $LEN(CVch(i))$  là hàm tìm số công việc trong bộ  $CVch(i)$  (bằng  $L$ ), và  $LEFMID(CVch(i),j)$  cho phép lấy công việc (số thứ tự) của công việc thứ  $j$ , phía bên trái trong bộ  $CVch(i)$ . Nếu tất cả các công việc của bộ này đã vẽ ( $X(l)=0$ , với mọi  $l$ ) thì sẽ thêm đỉnh  $k$  là đỉnh kết thúc của chúng ( $dcCV(l) = k$ ), và vẽ công việc  $i$  đi ra từ  $k$  ( $ddCV(i) = k$ ) (hình 3.8b).



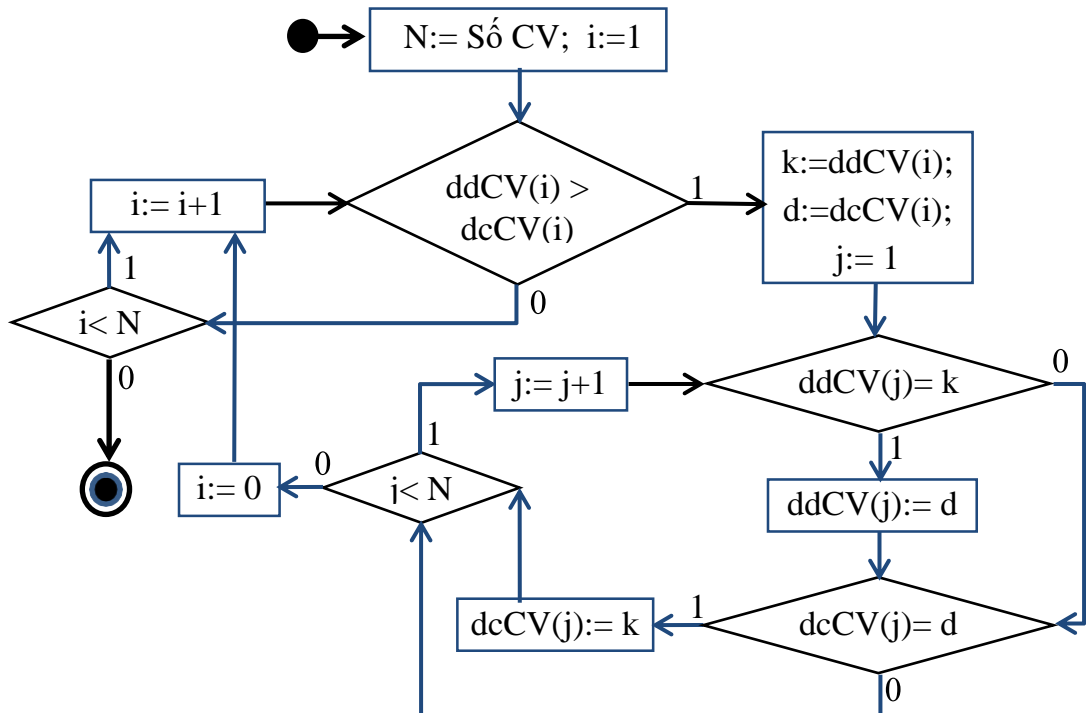
Hình 3.8b: Thêm đỉnh trung gian  $k$  và vẽ công việc đi ra từ  $k$

**Bước 3:** Chụm các công việc chưa có đỉnh kết thúc vào đỉnh cuối cùng được thêm vào làm đỉnh kết thúc dự án (hình 3.9).



Hình 3.9: Sơ đồ thuật toán vẽ đỉnh kết thúc mạng

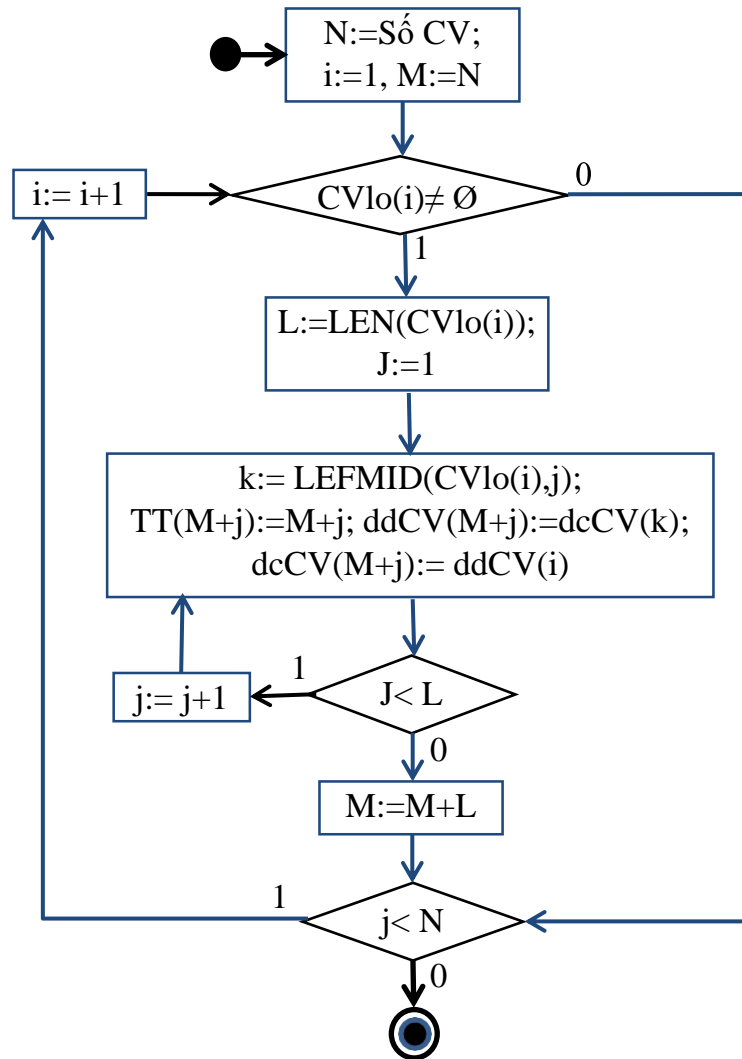
**Bước 4:** Đánh số lại các đỉnh, đảm bảo số đỉnh cuối của công việc phải lớn hơn số đỉnh đầu của công việc (hình 3.10).



Hình 3.10: Sơ đồ thuật toán đánh số lại các đỉnh của mạng

**Bước 5:** Thêm các công việc giả, để đảm bảo rằng buộc: mọi công việc đều đi sau các công việc đi trước nó đã cho ở cột “Công việc đi trước”. Cụ thể là, trong các

công việc đi trước, có công việc bị xóa theo bước 3 của thuật toán xác định đỉnh trung gian mục 3.3.1. ở trên. Do đó, công việc được vẽ không đi sau công việc bị xóa này. Vì vậy, mỗi công việc bị xóa cần thêm một công việc giả để phục hồi yêu cầu về các công việc đi trước một công việc đã cho (hình 3.11).



Hình 3.11: Sơ đồ thuật toán thêm các công việc giả

**c. Đánh giá độ phức tạp của thuật toán lập mạng AOA**

Gọi  $T_1(n)$ ,  $T_2(n)$  lần lượt là thời gian thực hiện của giai đoạn 1 và giai đoạn 2.

Trong giai đoạn 1 được chia ra là 2 bước bao gồm: bước đánh dấu các bộ công việc và bước xóa các bộ công việc.

Gọi bước 1 và bước 2 của giai đoạn (1) xác định đỉnh trung gian có thời gian thực hiện lần lượt là  $T_1B_1(n)$  và  $T_1B_2(n)$ . Dựa vào sơ đồ logic ta thấy  $T_1B_1(n) = O(n^2)$  và  $T_1B_2(n) = O(n^2)$ . Do đó  $T_1(n) = T_1B_1(n) + T_1B_2(n) = O(n^2)$ .

Tương tự như vậy, ta thấy với các bước của giai đoạn vẽ sơ đồ mạng có max thời gian thực hiện là  $O(n^2)$ . Vì vậy độ phức tạp của toàn thuật toán là  $O(n^2)$ .

### 3.2.4. Sơ đồ logic tính các tham số thời gian trên mạng AOA

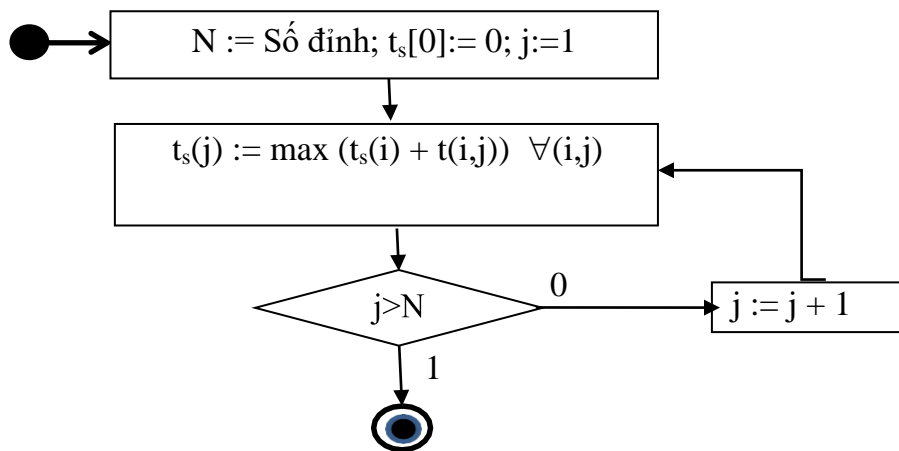
Dựa vào bảng sơ đồ mạng chúng ta có thể biết được các công việc bắt đầu từ đỉnh nào, kết thúc tại đỉnh nào. Từ đó có thể tính được các tham số thời gian như sau:

#### a. Tính thời gian bắt đầu sớm nhất của một đỉnh

Quá trình tính toán được thực hiện theo chiều xuôi, tức là lần lượt từ đỉnh 0 đến đỉnh N của sơ đồ mạng. Ta gọi:

- $t_s(j)$ : là thời gian bắt đầu sớm nhất của đỉnh j.
- $t(i,j)$ : là thời gian thực hiện công việc đi từ đỉnh i đến đỉnh j.

Sơ đồ logic để tính thời gian sớm nhất của các đỉnh của sơ đồ cho ở hình 3.12, được tiến hành trực tiếp trên bảng có các công việc mà đỉnh đầu và đỉnh cuối của chúng đã được xác định, thời gian thực hiện công việc đã cho.



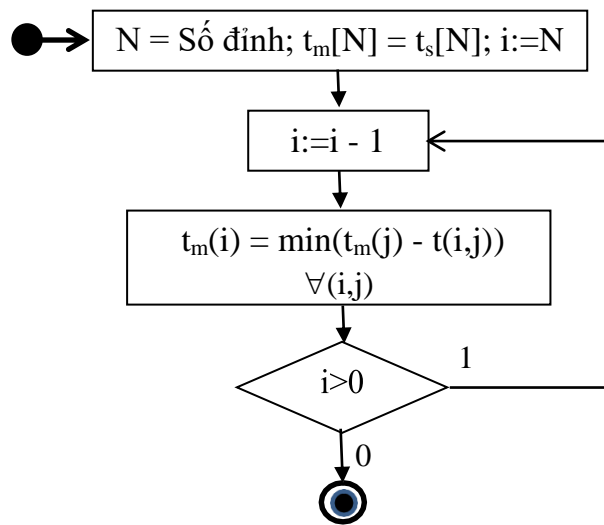
Hình 3.12: Sơ đồ logic tính thời gian bắt đầu sớm nhất của một đỉnh

**b. Tính thời gian kết thúc muộn nhất của một đỉnh**

Quá trình tính toán được thực hiện theo chiều ngược, lần lượt từ đỉnh N đến đỉnh 0 của sơ đồ mạng. Ta gọi:

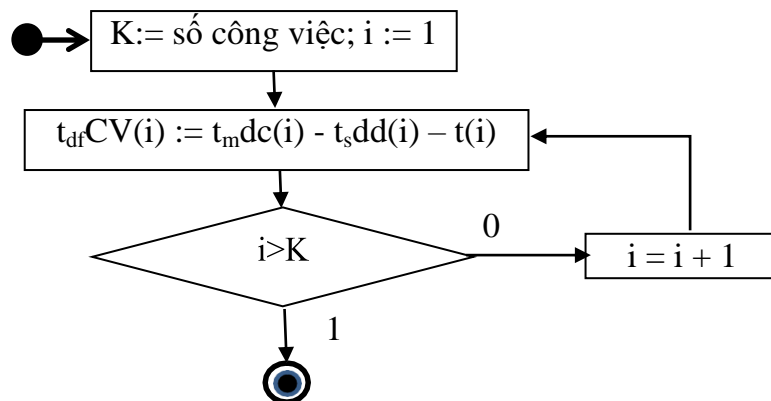
- $t_m(i)$ : là thời gian kết thúc muộn nhất của đỉnh i.
- $t(i,j)$ : là thời gian thực hiện công việc đi từ đỉnh i đến đỉnh j

Sơ đồ logic để tính thời gian kết thúc muộn nhất của các đỉnh của sơ đồ cho ở hình 3.13. (được thực hiện trên bảng).



Hình 3.13: Sơ đồ logic tính thời gian kết thúc muộn nhất của các đỉnh

**c. Tính thời gian dự phòng của công việc**



Hình 3.14: Sơ đồ logic tính thời gian dự phòng của công việc

Gọi  $t_{df}CV(i)$  là thời gian dự phòng của công việc  $i$ . Ta có sơ đồ tính toán cho ở hình 3.14. được tiến hành trực tiếp trên bảng lần lượt cho từng công việc một.

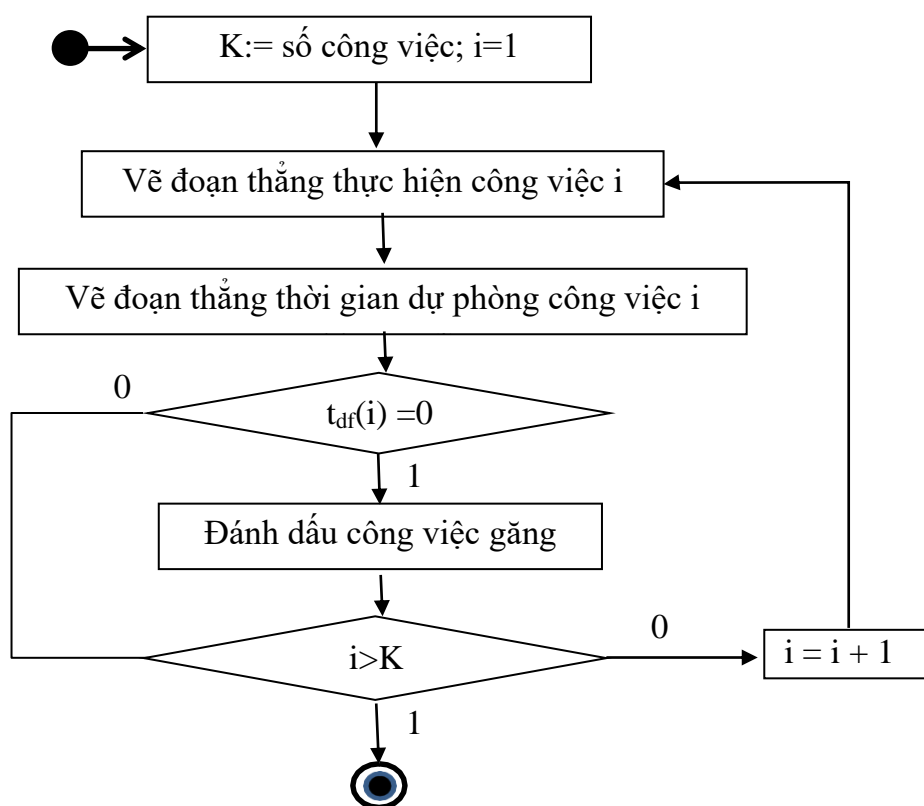
**d. Xác định các đỉnh và công việc thuộc đường găng**

Theo định nghĩa, đỉnh  $i$  là đỉnh găng nếu ta có  $t_m(i) = t_s(i)$  và công việc  $i$  được gọi là công việc găng nếu  $t_{df}(i) = 0$ . Khi kiểm tra các điều kiện này, ta dễ dàng đánh dấu được đỉnh găng và công việc găng.

**3.2.5. Sơ đồ logic vẽ các biểu đồ của kế hoạch lịch**

**a. Biểu đồ Gantt cho kế hoạch thời gian của dự án**

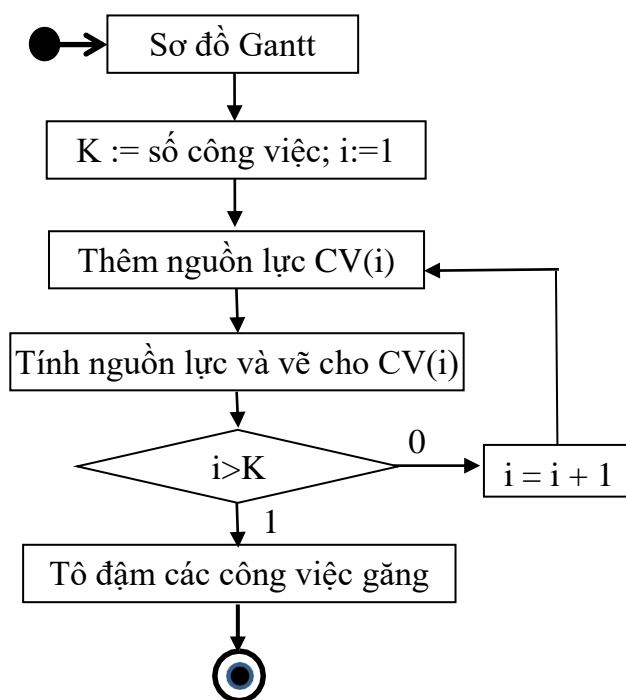
Vẽ biểu đồ Gantt của các công việc được thực hiện bằng cách vẽ đoạn thẳng tương ứng với mỗi công việc, và có độ dài bằng thời gian thực hiện công việc. Sau đó vẽ đoạn kéo dài của đoạn thẳng này có độ dài bằng thời gian dự phòng của công việc. Sơ đồ logic vẽ biểu đồ Gantt của kế hoạch dự án cho ở hình 3.15



Hình 3.15: Sơ đồ logic vẽ biểu đồ Gantt của kế hoạch dự án

### ***b. Biểu đồ sử dụng nguồn lực thực hiện dự án***

Dựa trên biểu đồ Gantt của kế hoạch dự án, ta có thể vẽ được sơ đồ sử dụng nguồn lực của dự án sau khi đã thêm các thông số về nguồn lực vào mỗi công việc. Sơ đồ vẽ biểu đồ sử dụng nguồn lực của kế hoạch dự án cho ở hình 3.16.



Hình 3.16: Sơ đồ logic vẽ biểu đồ sử dụng nguồn lực của dự án.

### **3.2.6. Giới thiệu về chương trình lập mạng cho kế hoạch lịch**

#### ***a. Môi trường và cấu trúc chương trình***

Chương trình lập mạng công việc và tính các tham số thời gian cho kế hoạch lịch trong quản lý dự án được viết bằng ngôn ngữ lập trình **Java** trên nền **web**, đồng thời sử dụng hệ quản trị cơ sở dữ liệu **MYSQL**.

Chương trình bao gồm 4 module như sau:

- Module xác định các đỉnh trung gian.
- Module xác định mạng công việc.
- Module tính các tham số về thời gian cho kế hoạch lịch

- Module vẽ biểu đồ Gantt cho kế hoạch lịch thời gian

*Dữ liệu đầu vào* của chương trình là một bảng phân rã các công việc bao gồm các thành phần: *Tên công việc, công việc trước, thời gian thực hiện công việc và nguồn lực sử dụng cho mỗi công việc.*

*Kết quả đầu ra* của chương trình bao gồm:

- *Một mạng công việc* trong đó các tham số về thời gian cho mỗi công việc. Mạng này có thể sử dụng cho việc lập mô hình cho các bài toán khác cần sử dụng đến mạng loại này. Chẳng hạn bài toán cân đối các nguồn lực cho một dự án.
- *Biểu đồ Gantt của kế hoạch lịch* giúp cho việc điều hành quản lý dự án.
- *Biểu đồ sử dụng nguồn lực* theo lịch thời gian (chưa được cân đối khi có tính đến sự hạn chế về nguồn lực) giúp việc điều phối nguồn lực.

### ***b. Hướng dẫn cài đặt***

Trước khi sử dụng phần mềm, cần phải cài đặt các phần mềm môi trường làm việc, bao gồm các phần mềm **Apache**, **MySQL** và **Tomcat**. Sau khi hoàn tất việc cài đặt và cấu hình các phần mềm trên, chúng ta có thể sử dụng các trình duyệt Internet như **Chrome**, **Firefox**, **Internet explorer** để bắt đầu sử dụng *chương trình “Chương trình lập lịch thời gian cho dự án đầu tư”*

### **3.2.7. Một số ví dụ thử nghiệm sử dụng chương trình thuật toán**

#### ***a. Ví dụ về lập mạng trên máy tính***



Bảng 3.3. Bảng phân rã chức năng công việc (đầu vào)

STT	Tên công việc	Công việc đi trước	Thời gian	Nhân lực
1	a		3	1
2	b		5	2
3	c		4	1
4	d	b	3	1
5	e	c	2	1
6	f	b,c	4	1
7	g	a	2	1
8	h	a,d	3	1
9	i	a,d	2	2
10	k	g	2	1
11	l	h,k	1	1

▶ Chạy dự án

↶ Quay lại

Thời gian chạy thuật toán: 0.00128 s

Bảng 3.4. Bảng xác định các đỉnh trung gian

STT	Tên công việc	Công việc được chọn	Công việc trước	Công việc loại
1	a			
2	b			
3	c			
4	d	b	b	
5	e	c	c	
6	f			b,c
7	g	a	a	
8	h	d	d	a
9	i	d	d	a
10	k	g	g	
11	l	h,k	h,k	

Bảng 3.5. Bảng tính các tham số thời gian các đỉnh của mạng

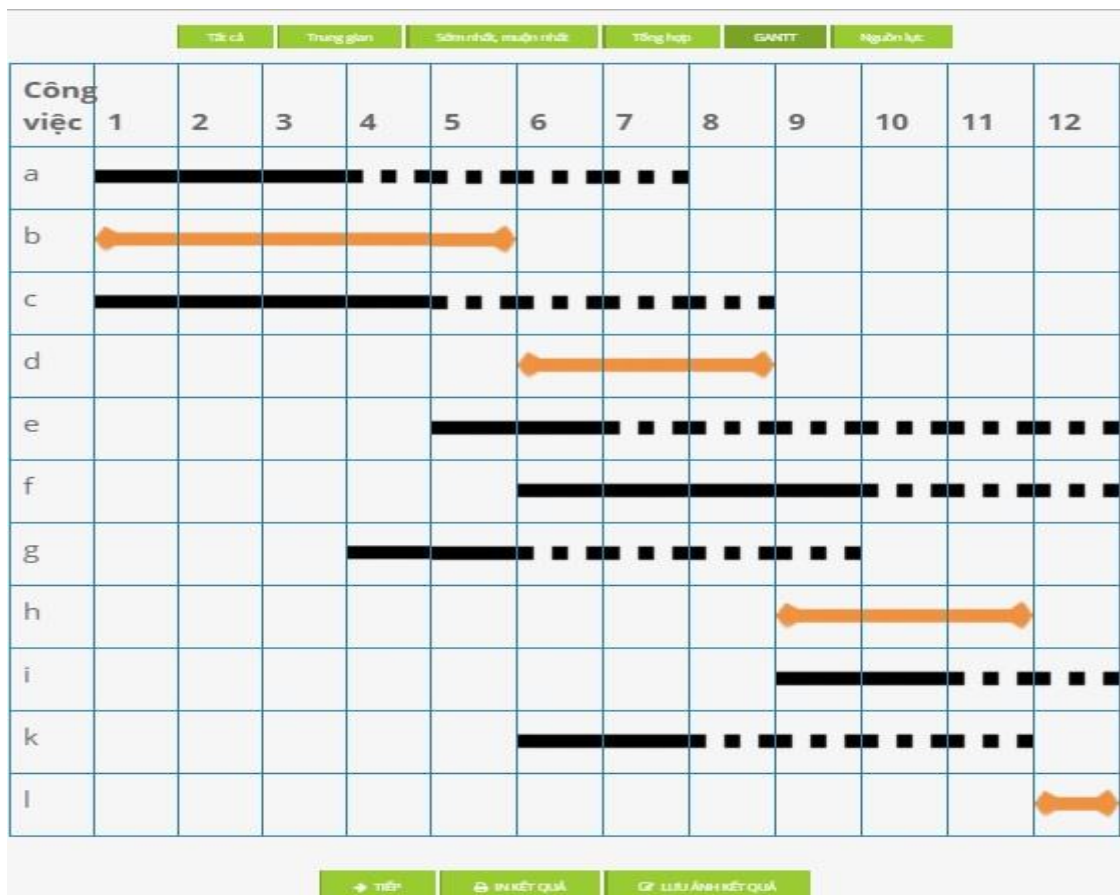
Tất cả					
Trung gian		Sớm nhất, muộn nhất		Tổng hợp	
Đỉnh	Thời gian bđ sớm nhất	Thời gian kt muộn nhất	Đỉnh găng		
0	0	0	✓		
1	5	5	✓		
2	4	8			
3	5	8			
4	3	7			
5	8	8	✓		
6	5	9			
7	11	11	✓		
8	12	12	✓		

[→ TIẾP](#)    [🖨 IN KẾT QUẢ](#)    [📷 LƯU ẢNH KẾT QUẢ](#)

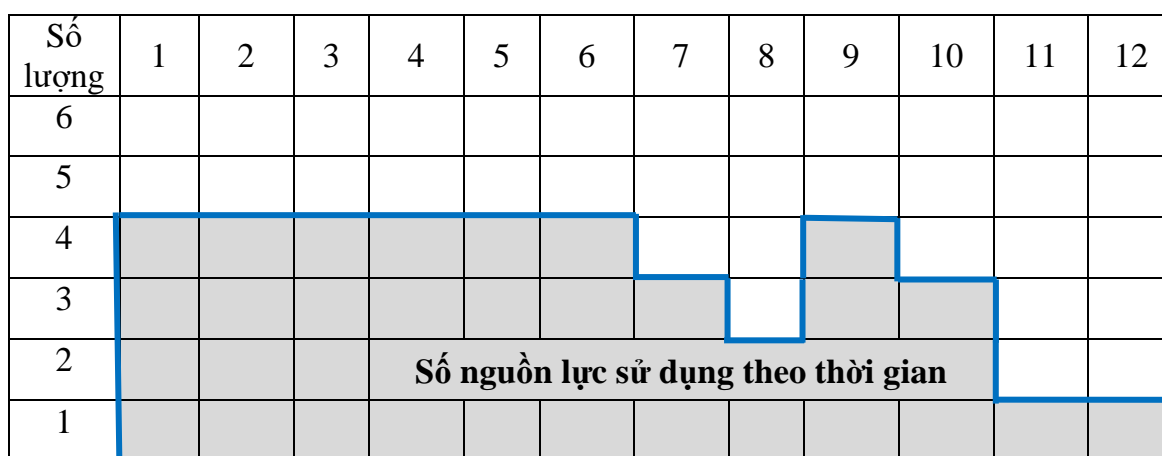
Bảng 3.6. Bảng lịch thực hiện công việc và đường găng

Tất cả							
Trung gian		Sớm nhất, muộn nhất		Tổng hợp		Nguồn lực	
STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	a		0	4	3	4	
2	b		0	1	5	0	✓
3	c		0	2	4	4	
4	d	b	1	5	3	0	✓
5	e	c	2	8	2	6	
6	f	b,c	3	8	4	3	
7	g	a	4	6	2	4	
8	h	a,d	5	7	3	0	✓
9	i	a,d	5	8	2	2	
10	k	g	6	7	2	4	
11	l	h,k	7	8	1	0	✓

[→ TIẾP](#)    [🖨 IN KẾT QUẢ](#)    [📷 LƯU ẢNH KẾT QUẢ](#)



Hình 3.17: Biểu đồ Gantt kế hoạch lịch của ví dụ



Hình 3.18: Biểu đồ sử dụng nguồn lực của ví dụ

***b. Tổng hợp kết quả thử nghiệm chương trình thuật toán***

Thuật toán đã chạy với 4 ví dụ khác nhau. Các kết quả chi tiết đưa ra từ chương trình có thể xem ở phụ lục 1. Bảng 3.7. là tổng hợp các kết quả thử nghiệm đã được chương trình thực hiện.

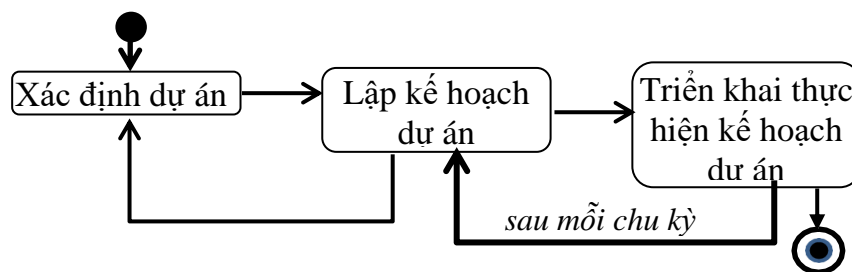
Bảng 3.7. Tổng hợp các kết quả thử nghiệm chương trình thuật toán

Số ví dụ	Số công việc	Số đỉnh của mạng	Tổng thời gian lập kế hoạch
1	10	6	0,00015 giây
2	10	6	0,00016 giây
3	10	5	0,00017 giây
4	18	10	0,76 giây
Tổng hợp	12	6	0,19012 giây

## Chương IV: XÂY DỰNG CHƯƠNG TRÌNH TRỢ GIÚP QUẢN LÝ DỰ ÁN

### 4.1. Bài toán quản lý, điều hành dự án và chương trình trợ giúp

Sau giai đoạn đầu xác định dự án (bao gồm có đánh giá dự án như đã trình bày ở trên), bước tiếp theo là quá trình triển khai thực hiện dự án. Ở giai đoạn này có hai hoạt động chính là *lập kế hoạch*, và *triển khai thực hiện kế hoạch*. Những công việc này được lặp lại sau mỗi chu kỳ nhất định cho đến khi kết thúc dự án (hình 4.1.)



Hình 4.1: Quá trình xác định và triển khai dự án

Thực sự thì ngay khi xác định dự án, không những cần đánh giá khả thi kinh tế (đã trình bày) mà còn phải lập kế hoạch để tiến hành đánh giá khả thi về mặt thời gian của dự án. Vì vậy, việc lập kế hoạch cho việc xác định dự án cũng được tiến hành một số lần trước khi dự án được chấp nhận triển khai. Trong quá trình triển khai thực hiện dự án, quá trình lập kế hoạch diễn ra thường xuyên sau mỗi chu kỳ thực hiện cho đến khi kết thúc (hình 4.1.).

Để đáp ứng mọi yêu cầu việc xác định, quản lý, điều hành dự án, chương trình phải trợ giúp được các hoạt động chung nhất thường gặp của quá trình này. Cụ thể:

#### 1. Lập một dự án mới

Mỗi khi lập một dự án mới, có hai hoạt động liên quan là: *tạo dự án mới* và *sửa đổi nội dung dự án* trong quá trình hình thành, do cần bổ sung hay thay đổi bảng công việc để đảm bảo dự án ngày một đầy đủ và chính xác hơn.

#### 2. Lập phiên bản mới cho dự án.

Thường với một dự án đặt ra, trong quá trình đàm phán hay lựa chọn phương thức thực hiện, có thể thay đổi cấu trúc trong chính dự án đó (thay đổi về công việc,

về quan hệ giữa chúng), cụ thể là chỉ phải thay đổi bảng phân rã công việc, nhưng không thay đổi nội phần chung của dự án (tên dự án, sản phẩm, các ràng buộc về thời gian và chi phí). Vì vậy phát sinh việc lập các phiên bản mới của dự án

### 3. Sửa dữ liệu cho một phiên bản đã có

Trong quá trình thực hiện dự án, cấu trúc dự án cơ bản không thay đổi, mà chỉ có dữ liệu cụ thể là thay đổi: ví dụ như công việc đã hoàn thành hay hoàn thành một phần. Khi đó dữ liệu về thời gian thực hiện và nguồn lực dành cho công việc đã thay đổi. Hoặc do ước lượng lại hay phân bổ lại nguồn lực cho việc thực hiện dự án ở giai đoạn tiếp theo cũng dẫn đến thay đổi dữ liệu của một phiên bản dữ liệu đã có. Vì thế cần có chức năng sửa dữ liệu của phiên bản.

### 4. Tiến hành lập mạng mới

Mỗi lần thay đổi cấu trúc dự án (tạo một phiên bản mới) ta cần phải lập lại mạng của dự án. Hoạt động này là một chức năng không thể thiếu theo cách lập kế hoạch của chúng ta.

### 5. Thực hiện tính toán và xây dựng kế hoạch lịch

Đây là nhiệm vụ chính của chương trình để trợ giúp cho quản lý dự án. Từ 5 nhiệm vụ trên đây sẽ là cơ sở để thiết kế hệ thống chương trình sau này.

## 4.2. Thiết kế dữ liệu vật lý cho chương trình

Cơ sở dữ liệu cho chương trình gồm các tệp dữ liệu sau đây:

### 1.tệp DUAN

Tên trường	Loại	Cỡ	Khóa	Ý nghĩa
maDA	varchar	8	Khóa chính	Mã dự án
tenDA	text	60		Tên dự án
nguoQL	text	30		Người quản lý
tongVon	int			Tổng vốn đầu tư
ngayBD	date			Ngày bắt đầu dự án
ngayKT	date			Ngày kết thúc dự án

## 2. tệp **PHIENBAN\_duan**

Tên trường	Loại	Cỡ	Khóa	Ý nghĩa
id	int	11	Khóa chính	Tên khóa
maDA	varchar	8		Mã dự án
phbanDA	varchar	4		
ngaysua	date			
ngayPB	date			
lichDA	varchar	256		
sodoNLDA	varchar	256		
daXuLy	bool	Yes/no		

## 3. tệp **DULIEU\_duan**

Tên trường	Loại	Cỡ	Khóa	Ý nghĩa
id	int	11	Khóa chính	Tên khóa
maDA	varchar	8		Mã dự án
phbanDA	varchar	4		Số phiên bản dự án (x.xx)
ngayPB	Date			Ngày bắt đầu phiên bản
thutuCV	Int	4		Số thứ tự công việc
tenCV	varchar	128		Tên công việc
cvDitruoc	text			Công việc đi trước
thoigianCV	Int	11		Thời gian thực hiện công việc
nguồnlucCV	Int	11		Nguồn lực thực hiện công việc

#### 4. tệpMANG\_da

Tên trường	Loại	Cỡ	Khóa	Ý nghĩa
id	Int	11	Khóa chính	Tên khóa
maDA	varchar	8	Khóa chính	Mã dự án
phbanDA	varchar	4		Số phiên bản dự án (x.xx)
ngaySua	Date	8		Ngày sửa dữ liệu phiên bản
thutuCV	Int	11		Số thứ tự công việc
tenCV	varchar	128		Tên công việc
dinhDau	int	4		Đỉnh đầu công việc
dinhCuoi	int	4		Đỉnh cuối công việc
thoigianCV	Int	11		Thời gian thực hiện công việc
thoigianBDCV	Int	11		Thời gian bắt đầu công việc
thoigianKTCV	Int	11		Thời gian kết thúc công việc
thoigianDFCV	Int	11		Thời gian dự phòng công việc
CVgang	tinyint	4		Công việc găng
nguồnlucCV	int	11		Nguồn lực thực hiện công việc

#### 5. tệp NGUOIDUNG\_da

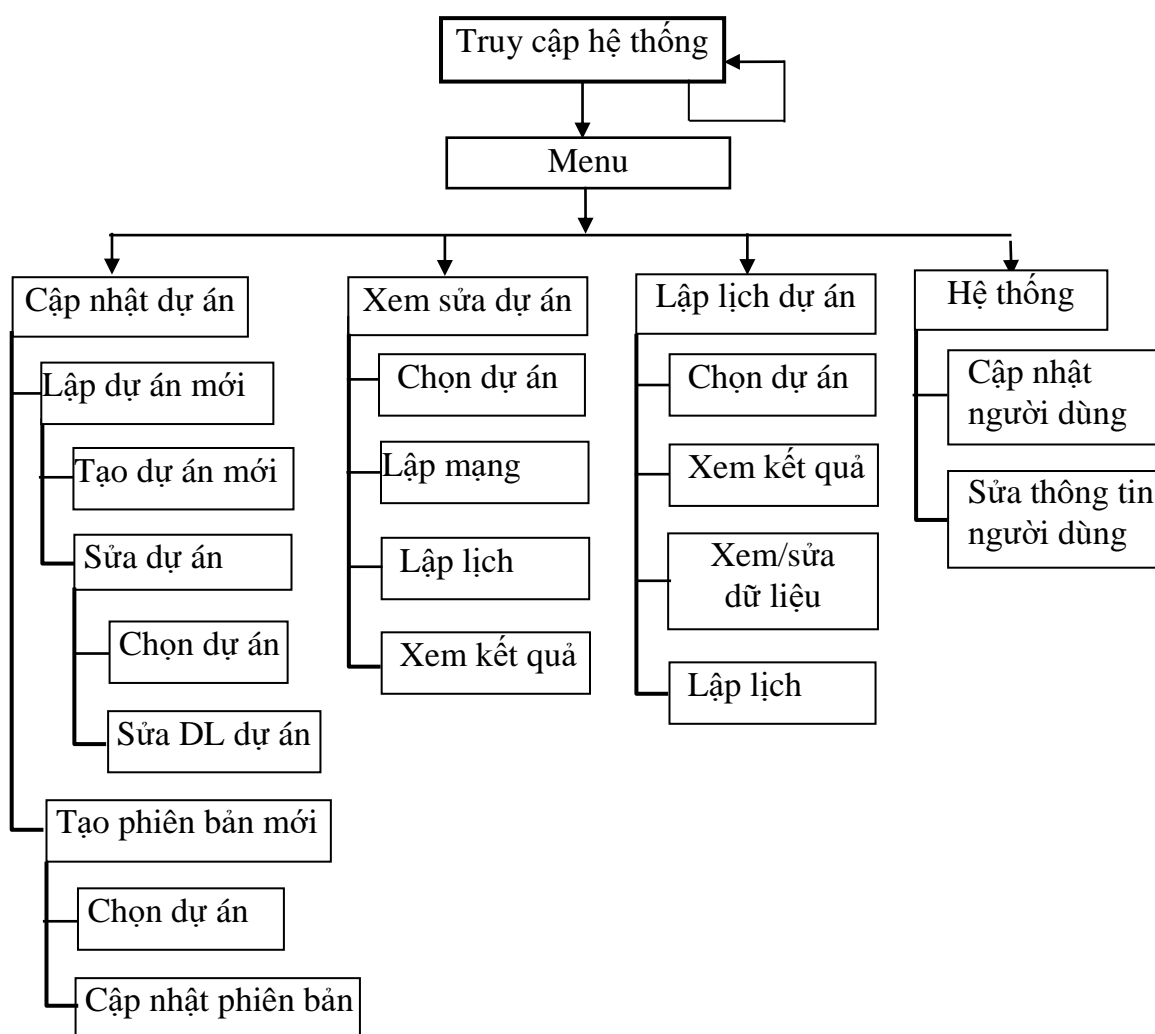
Tên trường	Loại	Cỡ	Khóa	Ý nghĩa
maNgdung	varchar	64	Khóa chính	Mã người dùng
tenNgdung	text			Tên người dùng
ngaysinhND	date			Ngày sinh người dùng
mailND	text			Mail người dùng
teleND	text			Điện thoại người dùng
passwordND	text			Mật khẩu người dùng
kichhoat	tinyint			Kích hoạt người dùng



### 4.3. Giới thiệu chương trình trợ giúp quản lý dự án

#### 4.3.1. Hệ thống thực đơn

Để trợ giúp được hoạt động lựa chọn và quản lý dự án, chương trình cần có các chức năng như đã nêu ở mục 4.1. Ngoài ra, để đáp ứng yêu cầu sử dụng chương trình của người dùng, hệ thống chương trình được thiết kế có cấu trúc như hình 4.2.[4]



Hình 4.2: Cấu trúc hệ thống thực đơn của chương trình

### 4.3.2. Một số chức năng chính của chương trình và giao diện

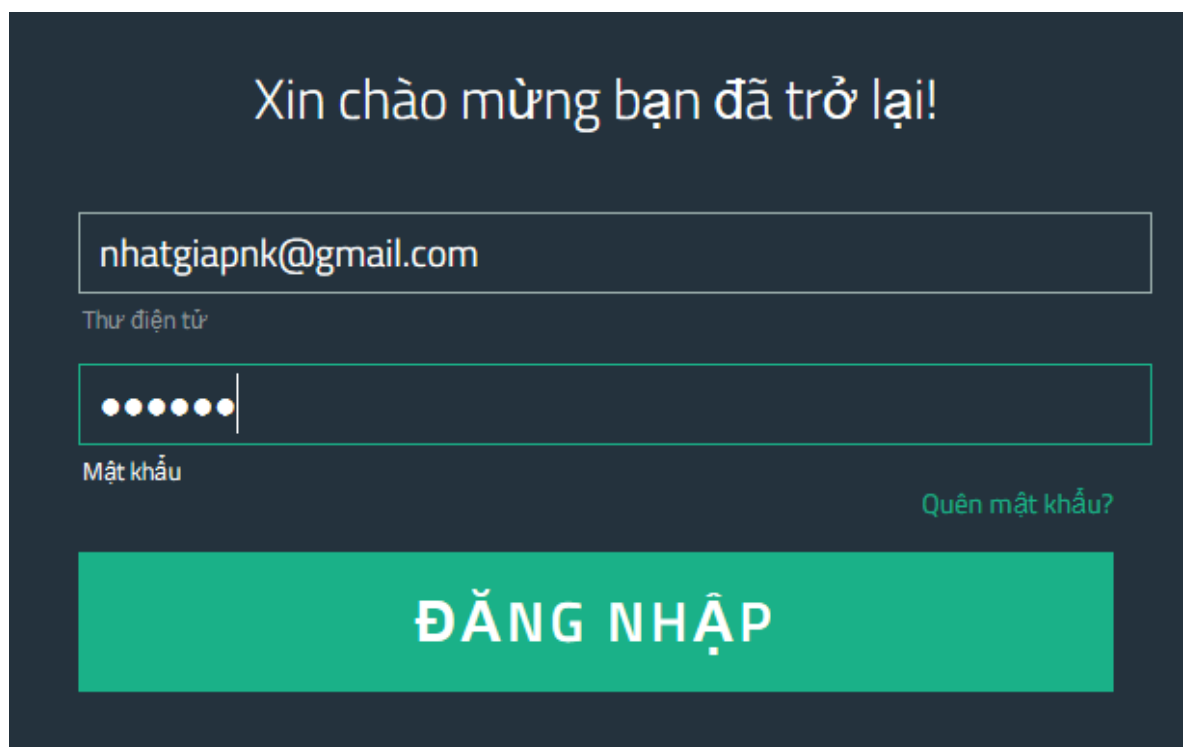
#### a. Đăng nhập hệ thống

Muốn sử dụng chương trình quản lý dự án cần thực hiện *đăng ký*. Người dùng cần đăng ký tài khoản để được đăng nhập vào hệ thống. Khi đăng ký tài khoản người dùng cần cung cấp họ và tên người dùng, địa chỉ email, mật khẩu đăng nhập.



The screenshot shows a dark-themed login page with the heading "CHÀO MỪNG BẠN ĐẾN VỚI CHƯƠNG TRÌNH". Below the heading are two input fields: "Thư điện tử\*" and "Mật khẩu\*". To the right of the password field is a link "Quên mật khẩu? | Đăng ký sử dụng". At the bottom is a large green button labeled "ĐĂNG NHẬP".

- *Khi đăng nhập hệ thống*, người dùng sử dụng thông tin đã đăng ký để đăng nhập hệ thống: Nhập địa chỉ email người dùng, mật khẩu người dùng



The screenshot shows a dark-themed login page with the heading "Xin chào mừng bạn đã trở lại!". Below the heading is an input field containing the email address "nhatgiapnk@gmail.com". Below the email field is the label "Thư điện tử". Below that is a password input field with five dots representing the password. Below the password field is the label "Mật khẩu". To the right of the password field is a link "Quên mật khẩu?". At the bottom is a large green button labeled "ĐĂNG NHẬP".

— Sau khi đăng nhập giao diện chính chương trình mở ra



### b. Cập nhật dự án



### c. Cập nhật phiên bản

#	Tên dự án	Người QL	Phiên bản	Ngày PB	Chức năng
VD1	Ví dụ 1	Trần nhật giáp	1.00	06/12/2016	Chi tiết
VD2	Ví dụ 2	Trần nhật giáp	1.00	06/12/2016	Chi tiết
VD3	Ví dụ 3	Trần nhật giáp	1.00	06/12/2016	Chi tiết
VD4	Ví dụ 4	Trần nhật giáp	1.00	06/12/2016	Chi tiết

#### d. Cập nhật dữ liệu

### DANH SÁCH CÁC CÔNG VIỆC TRONG DỰ ÁN

Các công việc trong dự án cần thực hiện đầu vào để lập mạng AOA

Các công việc trong dự án

Không có tệp nào được chọn

STT	Tên công việc	Công việc trước	Thời gian	Nhân lực	Chức năng
1	a		3	1	<input type="button" value="Sửa"/> <input type="button" value="Xóa"/>
2	b		5	2	<input type="button" value="Sửa"/> <input type="button" value="Xóa"/>
3	c		4	1	<input type="button" value="Sửa"/> <input type="button" value="Xóa"/>
4	d	b	3	1	<input type="button" value="Sửa"/> <input type="button" value="Xóa"/>
5	e	c	2	1	<input type="button" value="Sửa"/> <input type="button" value="Xóa"/>

#### e. Xem kết quả đã có

The screenshot shows the Engage web application interface. At the top, there is a navigation menu with the following items: **TRANG CHỦ**, **CẬP NHẬT DỰ ÁN**, **LẬP LỊCH DỰ ÁN**, **HỆ THỐNG**, and **ĐĂNG XUẤT**. Below the navigation menu is a large banner with the text "Cân Đẳng giảm lực" and "Sử dụng thuật toán đơn giản, hiệu quả lập mạng công việc". A dropdown menu is open, showing options: "Danh sách dự án", "Lập dự án mới", and "Tạo phiên bản mới". At the bottom of the banner, there is a URL: "ocalhost:8080/AONetworkAppv2.0/home#". Below the banner is a green bar with the text "Dự Án".

### DANH SÁCH CÁC DỰ ÁN

Các dự án sẽ thực hiện vẽ biểu đồ AOA

#	Tên dự án	Người QL	Tổng vốn	Ngày BD	Ngày KT	Chức năng
VD1	Ví dụ 1	Trần nhật giáp	200000000	01/12/2016	30/05/2016	<input type="button" value="Sửa"/> <input type="button" value="Phiên bản"/> <input type="button" value="Xóa"/>
VD2	Ví dụ 2	Trần nhật giáp	1000000000	02/12/2016	24/12/2016	<input type="button" value="Sửa"/> <input type="button" value="Phiên bản"/> <input type="button" value="Xóa"/>
VD3	Ví dụ 3	Trần nhật giáp	200000000	01/12/2016	30/12/2016	<input type="button" value="Sửa"/> <input type="button" value="Phiên bản"/> <input type="button" value="Xóa"/>
VD4	Ví dụ 4	Trần nhật giáp	200000000	02/12/2016	26/12/2016	<input type="button" value="Sửa"/> <input type="button" value="Phiên bản"/> <input type="button" value="Xóa"/>

## + Lập lịch dự án



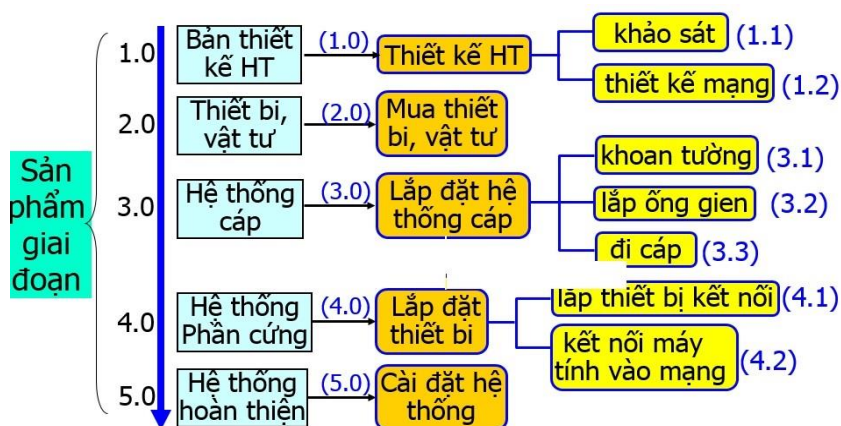
## + Hệ thống



### 4.3.3. Một ví dụ thực hiện dự án cụ thể với chương trình

#### a. Bài toán lắp đặt hệ thống mạng

Sơ đồ phân tích để xác định các công việc của bài toán được cho ở hình 4.3



Hình 4.3: Sơ đồ phân tích xác định công việc

Từ sơ đồ phân tích hình 4.3 [11]. Ta xác định được công việc cho bảng 4.0 và ước lượng thời gian thực hiện mỗi công việc cũng như số lượng nhân lực để thực hiện chúng.

Bảng 4.0. Bảng phân rã công việc của bài toán

Mã CV	Tên công việc	Công việc trước	Thời gian	Nhân lực
11	Khảo sát hệ thống	-	1	2
12	Lập thiết kế hệ thống	11	1	2
20	Mua sắm thiết bị	12	1	1
31	Khoan tường	12	2	1
32	Lắp ống gen	31	2	2
33	Đi cáp	32	1	2
41	Lắp các hộp kết nối	33	1	1
42	Kết nối máy tính vào mạng	41	1	2
50	Cài đặt, kiểm tra mạng	42	2	1

**b. Bảng dữ liệu đầu vào, phiên bản 1.00**

Từ sơ đồ hình 4.3. ta thiết lập được bảng dữ liệu phân rã chức năng của bài toán cho ở bảng 4.1. được in ra từ chương trình sau khi đã cập nhật dữ liệu bằng chức năng “ Tạo dự án mới”.

Bảng 4.1. Bảng phân rã chức năng của bài toán-phiên bản 1.00

Mã CV	Công việc trước	Thời gian	Nhân lực
11	-	1	2
12	11	1	2
20	12	1	1
31	12	2	1
32	31	2	2
33	32	1	2
41	33	1	1
42	41	1	2
50	42	2	1

**c. Các kết quả đưa ra của chương trình với phiên bản 1.00**

Khi chạy chương trình với chức năng “Lập mạng” ta được kết quả cho ở các bảng và hình sau:

Bảng 4.2. Bảng tính toán trung gian của bài toán-phiên bản 1.00

STT	Tên công việc	Công việc đi trước	Công việc được chọn	Công việc loại
1	11			
2	12	11	11	
3	20	12	12	
4	31	12	12	
5	32	31	31	
6	33	32	32	
7	41	33	33	
8	42	41	41	
9	50	42	42	

Bảng 4.3. Bảng thời gian bắt đầu, kết thúc của các đỉnh-phiên bản 1.00

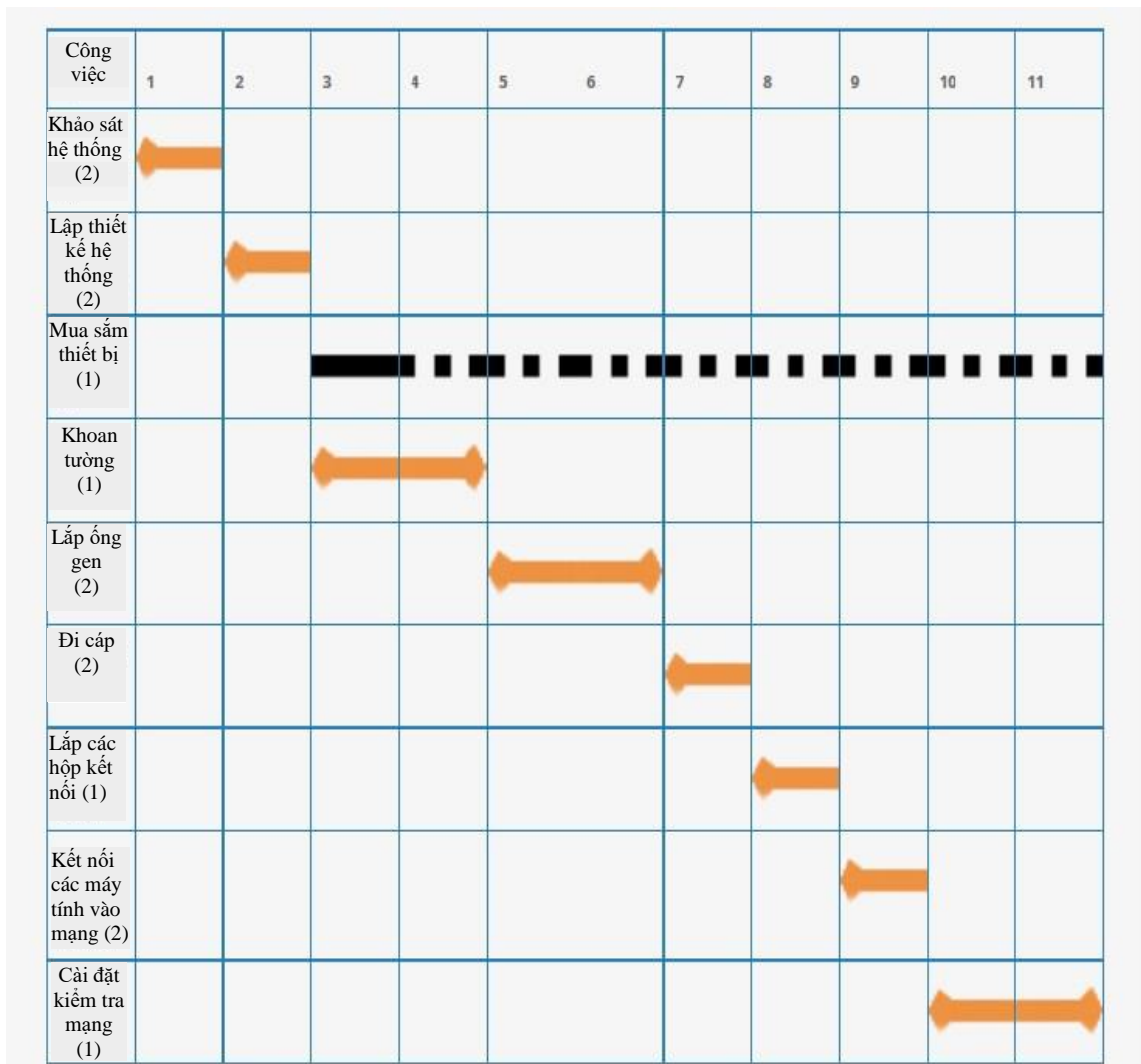
<span>Tất cả</span>   <span>Trung gian</span>   <span>Sớm nhất, muộn nhất</span>   <span>Tổng hợp</span>   <span>GANIT</span>   <span>Nguồn lực</span>			
Đỉnh	Thời gian bđ sớm nhất	Thời gian kt muộn nhất	Đỉnh găng
0	0	0	✓
1	1	1	✓
2	2	2	✓
3	4	4	✓
4	6	6	✓
5	7	7	✓
6	8	8	✓
7	9	9	✓
8	11	11	✓

→ TIẾP | ↩ IN KẾT QUẢ | 📄 Lưu Ám KẾT QUẢ

Bảng 4.4. Bảng lịch thực hiện công việc của bài toán-phiên bản 1.00

<span>Tất cả</span>   <span>Trung gian</span>   <span>Sớm nhất, muộn nhất</span>   <span>Tổng hợp</span>   <span>GANIT</span>   <span>Nguồn lực</span>							
STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	11		0	1	1	0	✓
2	12	11	1	2	1	0	✓
3	20	12	2	8	1	8	
4	31	12	2	3	2	0	✓
5	32	31	3	4	2	0	✓
6	33	32	4	5	1	0	✓
7	41	33	5	6	1	0	✓
8	42	41	6	7	1	0	✓
9	50	42	7	8	2	0	✓





Hình 4.4: Biểu đồ Gantt của lịch biểu kế hoạch của bài toán – phiên bản 1.00



Hình 4.5: Biểu đồ sử dụng nguồn lực lịch biểu của bài toán – phiên bản 1.00

Thời gian chạy thuật toán 0,00884 giây

**d. Bảng dữ liệu đầu vào Phiên bản 1.01: thay đổi thời gian và nhân lực**

Bảng 4.5. Bảng phân rã chức năng của bài toán - phiên bản 1.01

Mã CV	Công việc trước	Thời gian	Nhân lực
11	-	1	1
12	11	1	1
20	12	1	2
31	12	1	1
32	31	1	2
33	32	2	2
41	33	1	1
42	41	1	1
50	42	1	1

**e. Kết quả chạy đưa ra của phiên bản 1.01**

Bảng 4.6. Bảng tính toán trung gian của bài toán-phiên bản 1.01

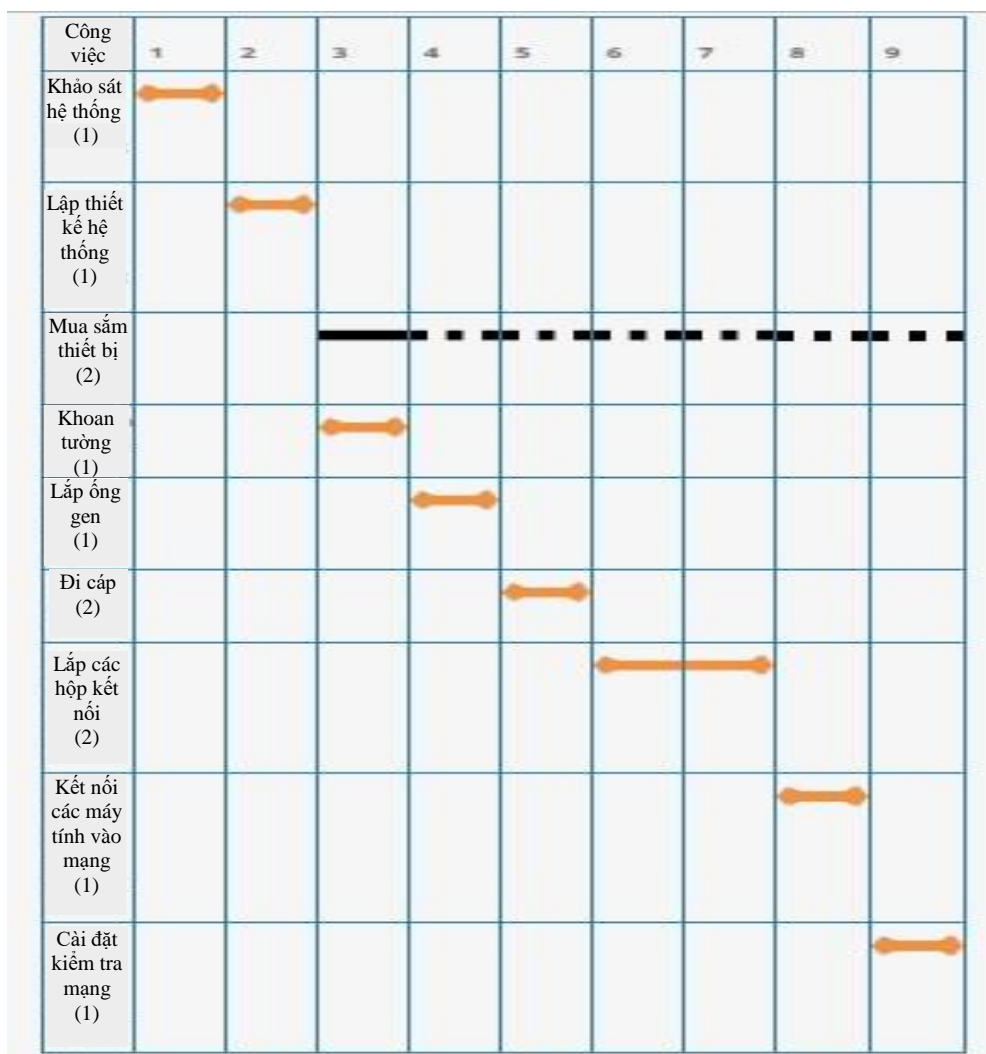
STT	Tên công việc	Công việc đi trước	Công việc được chọn	Công việc loại
1	11			
2	12	11	11	
3	20	12	12	
4	31	12	12	
5	32	31	31	
6	33	32	32	
7	41	33	33	
8	42	41	41	
9	50	42	42	

Bảng 4.7. Bảng thời gian bắt đầu, kết thúc của các đỉnh - phiên bản 1.01

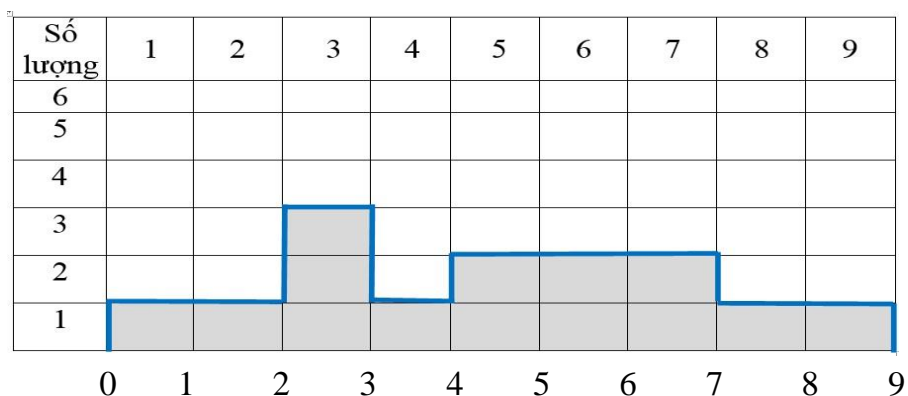
Đỉnh	Thời gian bđ sớm nhất	Thời gian kt muộn nhất	Đỉnh găng
0	0	0	✓
1	1	1	✓
2	2	2	✓
3	3	3	✓
4	4	4	✓
5	5	5	✓
6	7	7	✓
7	8	8	✓
8	9	9	✓

Bảng 4.8. Bảng lịch thực hiện công việc - phiên bản 1.01

STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	11		0	3	1	3	
2	12		0	3	1	3	
3	20		0	3	1	3	
4	31		0	3	1	3	
5	32		0	3	1	3	
6	33		0	1	2	0	✓
7	41	33	1	2	1	0	✓
8	42	41	2	3	1	0	✓
9	50		0	3	1	3	



Hình 4.6: Biểu đồ Gantt của lịch biểu kế hoạch của bài toán – phiên bản 1.01



Hình 4.7: Biểu đồ sử dụng nguồn lực của lịch biểu bài toán – phiên bản 1.01

Thời gian chạy thuật toán: 0.00024 giây

## KẾT LUẬN

Luận văn đã thực hiện được những công việc sau:

### *a. Về mặt lý thuyết*

- Luận văn đã nói về các vấn đề chung của quản lý dự án, đặc biệt là các khó khăn đang gặp phải tại thời điểm hiện tại. Giúp chúng ta hiểu được một trong những khó khăn của quản lý dự án là vấn đề lập lịch trên cơ sở cân đối nguồn nhân lực bị hạn chế.
- Nghiên cứu về thuật toán lập mạng AOA, một trong những thuật toán phức tạp trong việc xây dựng mạng. Dựa vào thuật toán này chúng ta hoàn toàn có thể vẽ được bằng tay mạng công việc. Các nhà quản lý dự án có thể hiểu và áp dụng đối với dự án của mình.
- Xây dựng thuật toán tự động hóa bài toán lập lịch biểu của dự án dựa trên mạng AOA, dựa vào việc xây dựng hóa này chúng ta sẽ giúp cho người dùng hiểu hơn về hệ thống, cũng như sự đúng đắn của quá trình xây dựng hệ thống dựa trên mạng AOA.

### *b. Về mặt ứng dụng*

- Giải quyết được bài toán lập lịch biểu trên cơ sở cân đối nguồn nhân lực bị hạn chế.
- Hệ thống cho phép tự động hóa thuật toán lập lịch trên mạng AOA trên nền web một cách nhanh chóng, tiện dụng và tăng tốc độ xử lý để phục hoạt động quản lý dự án lên nhiều lần.

### *c. Khả năng ứng dụng trong thực tiễn:*

- Giải quyết được bài toán lập lịch biểu trên cơ sở cân đối nguồn nhân lực bị hạn chế.
- Hệ thống cho phép tự động hóa thuật toán lập lịch trên mạng AOA trên nền web một cách nhanh chóng, tiện dụng và tăng tốc độ xử lý để phục hoạt động quản lý dự án lên nhiều lần.

#### ***d. Những hạn chế và tồn tại***

Mặc dù đạt được một số ưu điểm như trên nhưng hệ thống vẫn không tránh khỏi một số hạn chế và tồn tại. Cụ thể đó là:

- Vấn đề bảo mật chưa thực sự tối ưu
- Vấn đề an toàn, mất mát dữ liệu khi hệ thống bị tấn công hoặc phá hoại

#### ***e. Những hướng nghiên cứu tiếp theo:***

Trong thời gian tới, luận văn tập trung các hướng nghiên cứu sau để làm tăng hiệu quả hệ thống:

- Tiếp tục nghiên cứu, thử nghiệm và hoàn thiện giải pháp bảo mật cho hệ thống.
- Nghiên cứu và áp dụng giải pháp phân tải tối ưu nhất cho hệ thống để hệ thống chạy hiệu quả và ổn định hơn khi có lượng người truy cập cao trong cùng một thời điểm.
- Nghiên cứu một vài mô hình tối ưu hóa tài nguyên cho bài toán lập lịch mà được nhiều tác giả nêu ra để đưa vào chương trình.
- Tiếp tục cải tiến và sử dụng các phương pháp sao lưu hiệu quả nhất để đảm bảo khi có sự cố, dữ liệu hệ thống được phục hồi nhanh chóng và đầy đủ.

## TÀI LIỆU THAM KHẢO

### Tiếng Việt

1. Lê Văn Kiêm, Ngô Quang Tương (2008), *Quản lý dự án bằng sơ đồ mạng*, NXB Xây dựng, tr. 13-20, Hà Nội.
2. Từ Quang Phương (2010), *Quản lý dự án*, NXB Đại học Kinh tế Quốc dân, Hà Nội.
3. Lê Văn Phùng (2014), *Kỹ nghệ phần mềm*, NXB Thông tin và truyền thông, Hà Nội.
4. Lê Văn Phùng (2014), *Các mô hình cơ bản trong phân tích và thiết kế hướng đối tượng*, NXB Thông tin và truyền thông, Hà Nội.
5. TS.Lê Văn Phùng, Ths.Trần Nguyên Hương, CN.Lê Hương Giang (2015), *Quản lý dự án CNTT*. Nhà xuất bản Thông tin và Truyền thông, Hà Nội.
6. Nguyễn Quang Thái, Hoàng Phong Oanh (1969), *Phương pháp sơ đồ mạng lưới*, NXB Khoa học và Kỹ thuật, Hà Nội.
7. Trịnh Quốc Thắng (2006), *Quản lý dự án xây dựng*, NXB Khoa học và Kỹ thuật, Hà nội.
8. Trịnh Quốc Thắng (1978), *Sơ đồ mạng trong xây dựng*, Trường Đại học Xây dựng Hà Nội.
9. Nguyễn Văn Vy, Nguyễn Việt Hà (2008), *Giáo trình kỹ nghệ phần mềm*, NXB Đại học QGHN.
10. Nguyễn Văn Vy, Trần Thị Thu Minh (2012), “Một thuật toán mới lập mạng công việc AOA (Activities On Arcs)”, *Tạp chí Kinh tế phát triển*, số 180.
11. Nguyễn Văn Vy (2013). Slides bài giảng quản lý dự án, Hà Nội.

### Tiếng Anh

12. Cohen, Yuval and Sadeh. Arik, 2007, “A New Approach for Constructing and Generating AOA Networks”, *Journal of Computer Science*, Volume 1, Issue 1.
13. David I. Cleland, Roland Gareis, 2006, *Global project management handbook*. McGraw-Hill Professional.
14. Dimsdale, D. *Computer construction of minimal project network*, IBM systems journal, 2,24-36, 1963
15. Hirsch, Georges and Đặng Hữu Đạo, Nguyễn Tiên Dũng, Nguyễn Chân , 1994, *Quản lý dự án*, NXB Giáo dục, Trung tâm Pháp – Việt về Quản lý, Hà Nội, trang: 188-209.
16. Howes, Norman R, 2001, *Modern Project Management. (Sucessfully Integrating Project Management Knowledge Areas and Process)*. AMCOM – American Management Association.

17. Hughes, Bob and Mike Cotterell, 2002,*Software Project Management*, Third Edition, McGraw-Hill.
18. Kamburowski, J. D., Michael, J. and Stallman, M. (1992). *Optimal construction of Project Activity Networks*, *Proceeding of the Annual Meeting of the Decision Sciences Institute*, San-Francisco, CA,1424-1426
19. Kamburowski, J. D., Michael, J. and Stallman, M, 2000,*Minimizing the Complexity of an Activity Network*, *Networks*, 36 (1), 47-52.
20. Kelley, James, Walker, Morgan, *Critical-Path Planning and Scheduling* (1959), Proceedings of the Eastern Joint Computer Conference.
21. Krishnamoorthy, M. S. and Deon,N, 1979,*Complexity of minimum-Dummy-Activities Problem in a PERT Network*, *Networks*, 9, 189-194.
22. Michael, D.; Kambourowski,J. and Stalim, M, 1993,*On the minimum dummy-arc problem*, *Recherche opérationnelle/Operations Research*, vol. 27, no 2, p. 153 2 168.
23. Mouhoub, N.E, Abdelhamid Benhocine, 2011,*An efficient algorithm for generating AoA network A networks*. Information System department, Qassim University, Saudi Arabia. Abdelhamid- [benhocine@yahoo.fr](mailto:benhocine@yahoo.fr).
24. Project Management Institute, 2004,*A Guide to the, Project Management Body of Knowledge*, 3rd Edition, ANSI/PMI.
25. Spinard, J, 1980, *The Minimum Dummy Task Problem*, *Networks*, 16, 331-348.
26. Syslo,M..M. (1981), *Optimal Construction of Event-Node Networks*, *RAIRO*, 15, 241-260.
27. Willis, R.J, 1985, An algorithm for constructing project network diagrams on an ordinary line printer. *Computers & Operations Research*, Volume 12, Issue 2, Pages 163-168.



## PHỤ LỤC

### A. PHỤ LỤC 1: Một số kết quả tính toán của chương trình lập lịch

#### A1.1. Ví dụ 1

Bảng A1.1. Bảng dữ liệu đầu vào ví dụ 1

STT	Tên công việc	Công việc đi trước	Thời gian	Nhân lực
1	a		3	1
2	b		5	2
3	c	a	4	1
4	d	a,b	3	1
5	e	b	2	1
6	f	b	4	1
7	g	e,d	2	1
8	h	e,d	3	1
9	i	c	2	2
10	k	c,h,f	2	1

▶ Chạy dự án      ↶ Quay lại

Thời gian chạy thuật toán: 0.00015 s

Bảng A1.2. Kết quả trung gian ví dụ 1

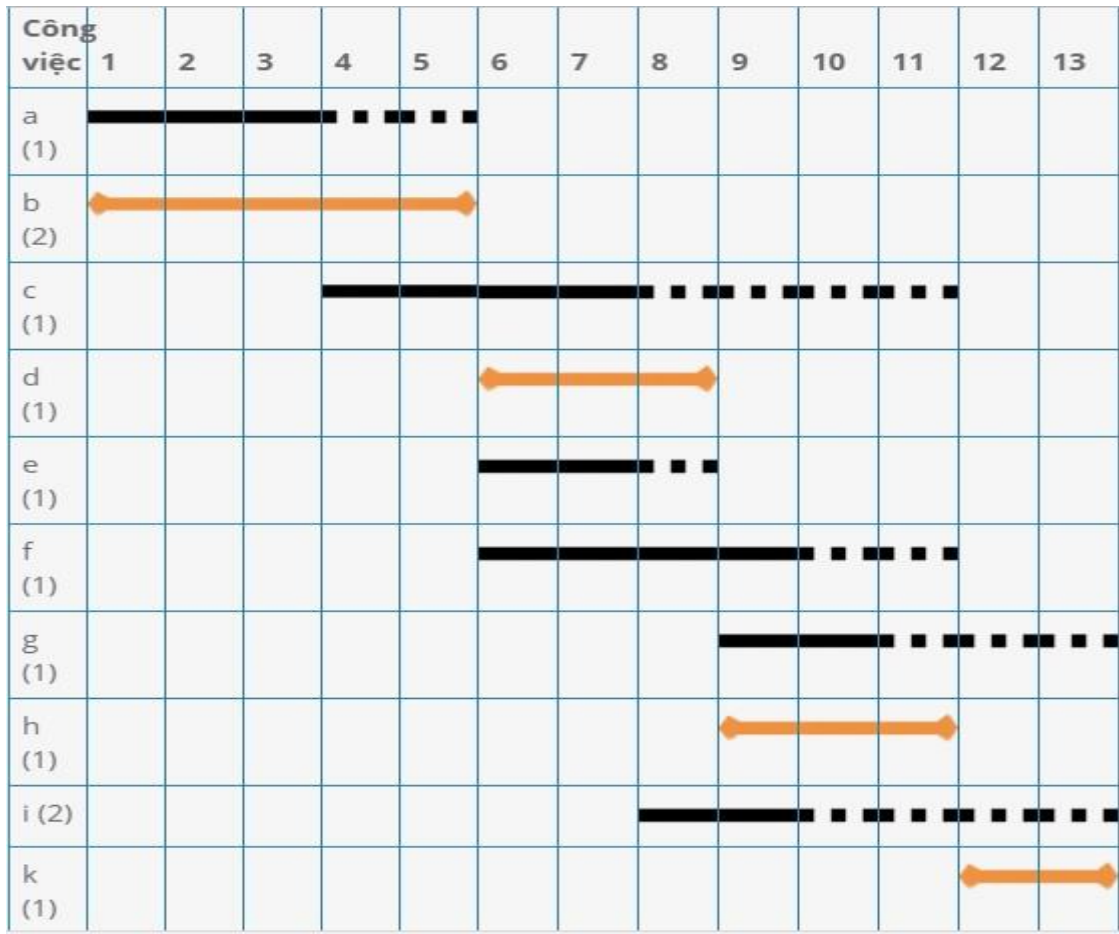
STT	Tên công việc	Công việc đi trước	Công việc được chọn	Công việc loại
1	a			
2	b			
3	c	a	a	
4	d	a,b		a,b
5	e	b	b	
6	f	b	b	
7	g	e,d	e,d	
8	h	e,d	e,d	
9	i	c	c	
10	k	c,h,f	h,f	c

Bảng A1.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 1

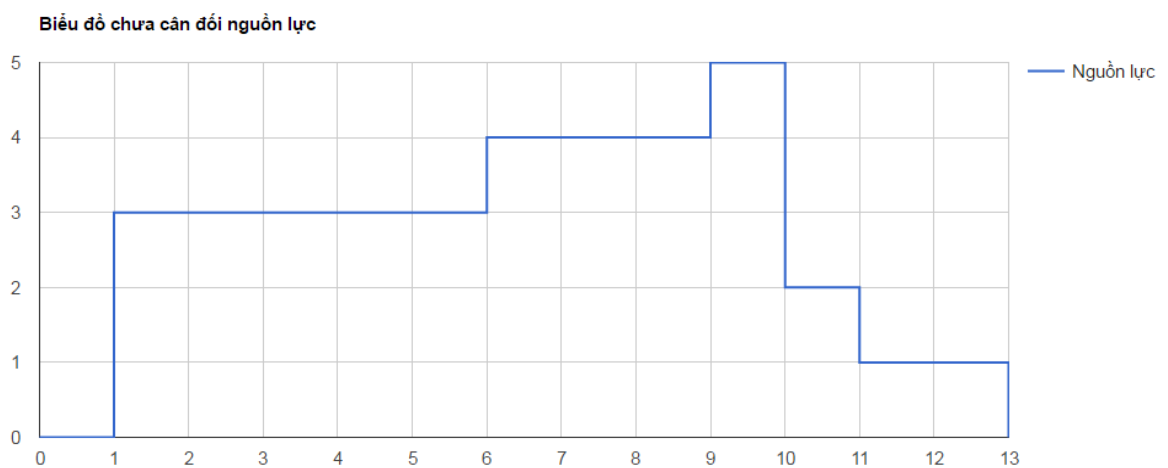
Đỉnh	Thời gian bđ sớm nhất	Thời gian kt muộn nhất	Đỉnh găng
0	0	0	✓
1	3	5	
2	5	5	✓
3	5	5	✓
4	8	8	✓
5	7	11	
6	11	11	✓
7	13	13	✓

Bảng A1.4. Bảng lịch thời gian thực hiện công việc của ví dụ 1

STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	a		0	1	3	2	
2	b		0	2	5	0	✓
3	c	a	1	5	4	4	
4	d	a,b	3	4	3	0	✓
5	e	b	2	4	2	1	
6	f	b	2	6	4	2	
7	g	e,d	4	7	2	3	
8	h	e,d	4	6	3	0	✓
9	i	c	5	7	2	4	
10	k	c,h,f	6	7	2	0	✓



Hình A1.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 1



Hình A1.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 1

Thời gian thực hiện chương trình: **0.00015 giây**

## A1.2. Ví dụ 2

Bảng A2.1. Bảng dữ liệu đầu vào ví dụ 2

STT	Tên công việc	Công việc đi trước	Thời gian	Nhân lực
1	a		3	1
2	b		4	1
3	c	a	5	1
4	d	a,b	6	1
5	e	a,b	5	1
6	f	b	4	1
7	g	c,d	2	1
8	h	e,f	3	1
9	i	e,f	2	1
10	k	c,d,h	1	1

▶ Chạy dự án      ◀ Quay lại

Thời gian chạy thuật toán: 0.00016 s

Bảng A2.2. Kết quả trung gian ví dụ 2

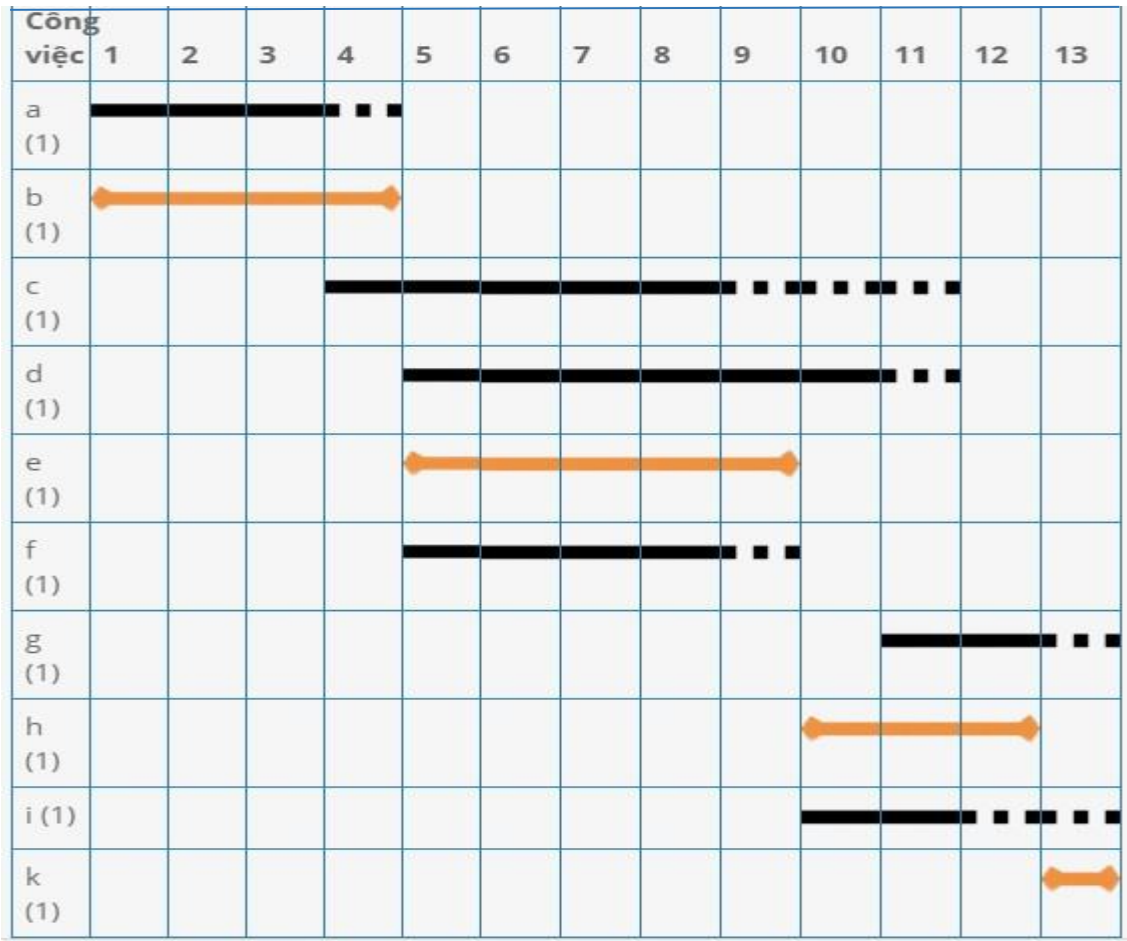
STT	Tên công việc	Công việc đi trước	Công việc được chọn	Công việc loại
1	a			
2	b			
3	c	a	a	
4	d	a,b		a,b
5	e	a,b		a,b
6	f	b	b	
7	g	c,d	c,d	
8	h	e,f	e,f	
9	i	e,f	e,f	
10	k	c,d,h	h	c,d

Bảng A2.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 2

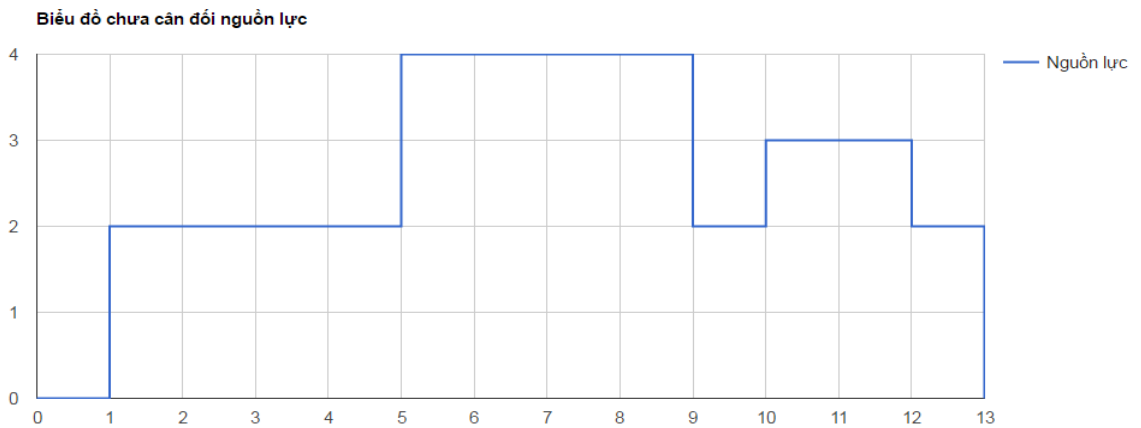
Đỉnh	Thời gian bắt sớm nhất	Thời gian kết muộn nhất	Đỉnh găng
0	0	0	✓
1	3	4	
2	4	4	✓
3	4	4	✓
4	4	5	
5	10	11	
6	9	9	✓
7	12	12	✓
8	13	13	✓

Bảng A2.4. Bảng lịch thời gian thực hiện công việc của ví dụ 2

STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	a		0	1	3	1	
2	b		0	2	4	0	✓
3	c	a	1	5	5	3	
4	d	a,b	4	5	6	1	
5	e	a,b	3	6	5	0	✓
6	f	b	2	6	4	1	
7	g	c,d	5	8	2	1	
8	h	e,f	6	7	3	0	✓
9	i	e,f	6	8	2	2	
10	k	c,d,h	7	8	1	0	✓



Hình A2.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 2



Hình A2.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 1

Thời gian chạy chương trình: **0.00016 giây**

### A1.3. Ví dụ 3

Bảng A3.1. Bảng dữ liệu đầu vào ví dụ 3

STT	Tên công việc	Công việc đi trước	Thời gian	Nhân lực
1	a		3	1
2	b		4	2
3	c	a	5	1
4	d	a,b	6	1
5	e	a,b	5	1
6	f	b	4	1
7	g	c,d	2	1
8	h	e,f	3	1
9	i	e,f	2	2
10	j	c,d,h	2	1

▶ Chạy dự án Quay lại

Thời gian chạy thuật toán: 0.00017 s

Bảng A3.2. Kết quả trung gian ví dụ 3

STT	Tên công việc	Công việc đi trước	Công việc được chọn	Công việc loại
1	a			
2	b			
3	c	a	a	
4	d	a,b		a,b
5	e	a,b		a,b
6	f	b	b	
7	g	c,d	c,d	
8	h	e,f	e,f	
9	i	e,f	e,f	
10	j	c,d,h	h	c,d

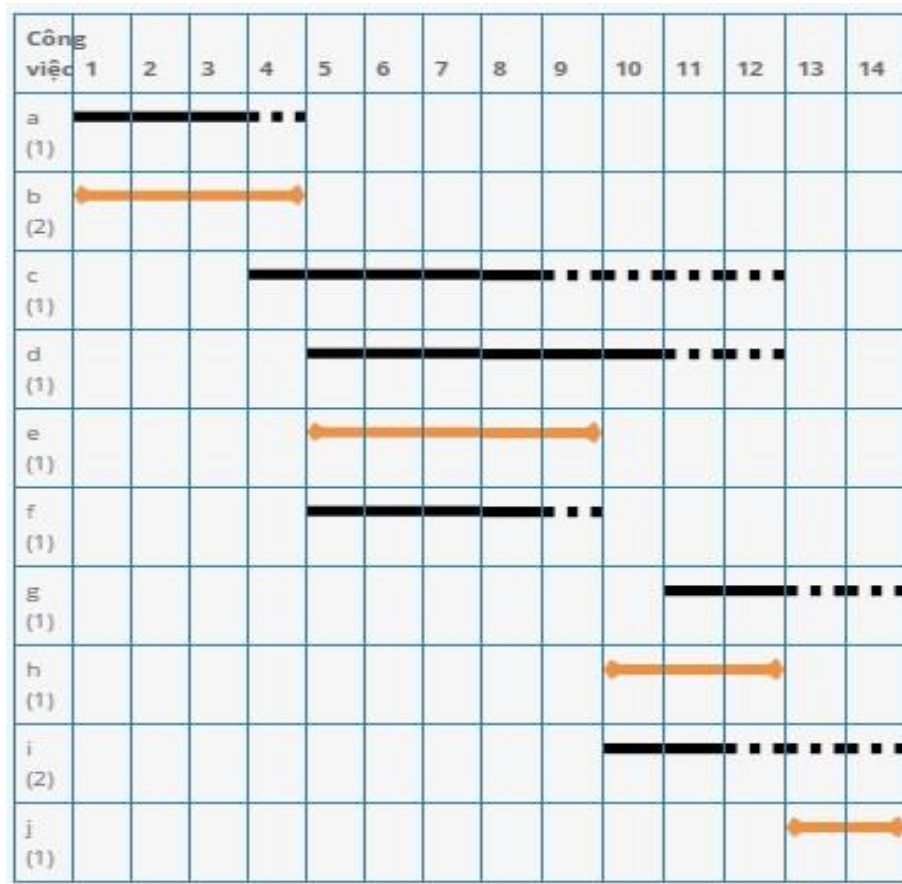
Bảng A3.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 3

Đỉnh	Thời gian bđ sớm nhất	Thời gian kt muộn nhất	Đỉnh găng
0	0	0	✓
1	3	6	
2	0	5	
3	0	3	
4	4	4	✓
5	8	11	
6	8	8	✓
7	11	11	✓
8	13	13	✓

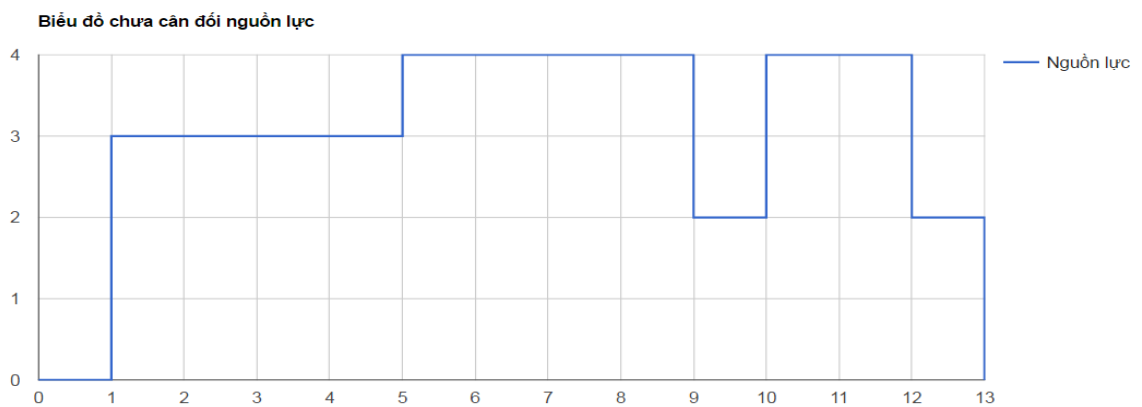
Bảng A3.4. Bảng lịch thời gian thực hiện công việc của ví dụ 3

STT	Tên công việc	Công việc đi trước	Đỉnh đầu	Đỉnh cuối	Thời gian thực hiện công việc	Tg dự phòng	Công việc găng
1	a		0	1	3	1	
2	b		0	2	4	0	✓
3	c	a	1	5	5	4	
4	d	a,b	4	5	6	2	
5	e	a,b	3	6	5	0	✓
6	f	b	2	6	4	1	
7	g	c,d	5	8	2	2	
8	h	e,f	6	7	3	0	✓
9	i	e,f	6	8	2	3	
10	j	c,d,h	7	8	2	0	✓





Hình A3.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 3



Hình A3.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 3

Thời gian thực hiện chương trình: **0.00017** giây

#### A1.4. Ví dụ 4

Bảng A4.1. Bảng dữ liệu đầu vào ví dụ 4

#	Tên công việc	Công việc trước	Thời gian	Nhân lực
1	a		1	1
2	b		5	1
3	c		6	1
4	d		4	1
5	e	a	4	1
6	f	b	3	1
7	g	c	3	1
8	h	d	4	1
9	i	e,f	2	1
10	k	g,i	2	1
11	m	i	6	1
12	l	i	3	1
13	n	k	2	1
14	o	l,n	1	1
15	p	g,h,i	2	1
16	q	g,h,i	3	1
17	r	o,p	2	1
18	s	q,r	1	1

Bảng A4.2. Kết quả trung gian ví dụ 4

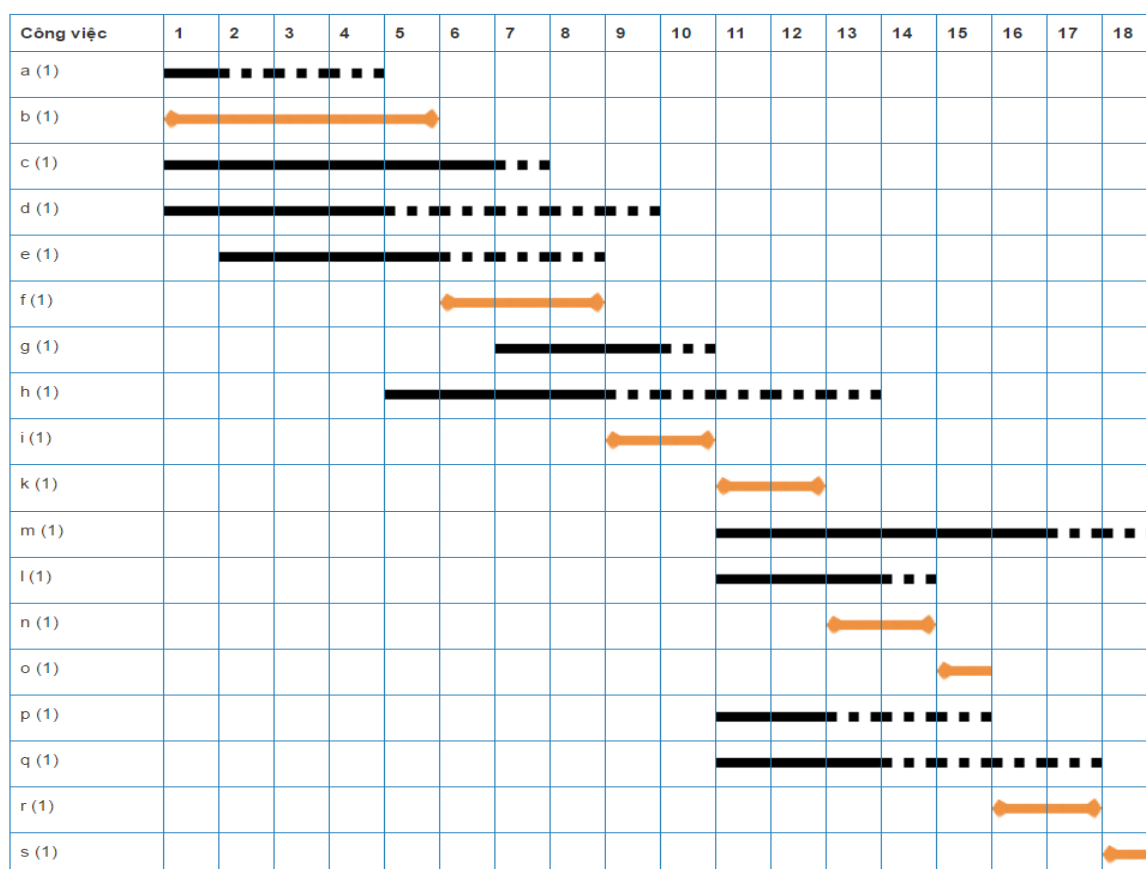
#	Tên công việc	Công việc trước	Công việc loại
1	a		
2	b		
3	c		
4	d		
5	e	a	
6	f	b	
7	g	c	
8	h	d	
9	i	e,f	
10	k	g	i
11	m	i	
12	l	i	
13	n	k	
14	o	l,n	
15	p	h	i,g
16	q	h	i,g
17	r	o,p	
18	s	q,r	

Bảng A4.3. Bảng thời gian bắt đầu và kết thúc của đỉnh, ví dụ 4

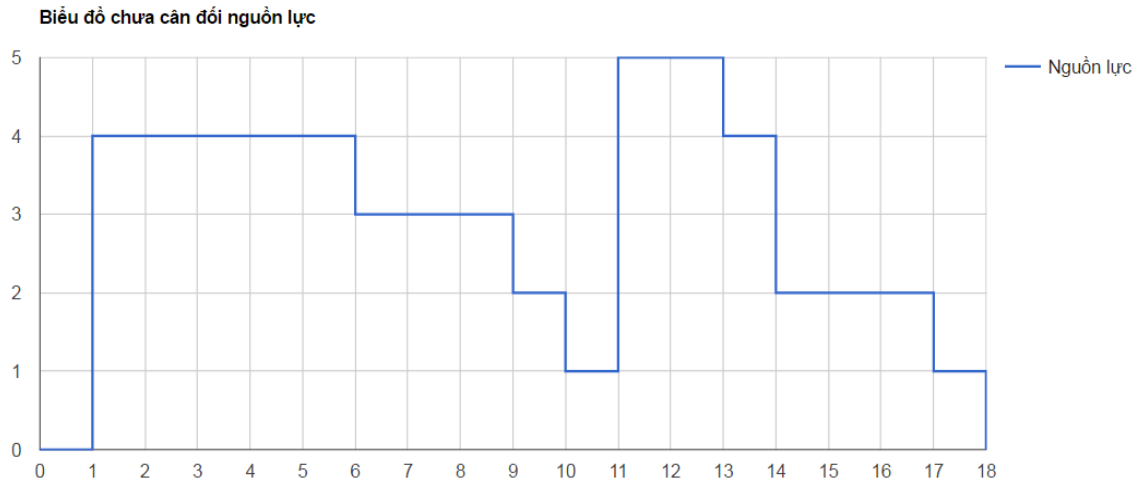
Đỉnh	Thời gian bắt đầu sớm nhất	Thời gian kết thúc muộn nhất	Đỉnh găng
0	0	0	✓
1	1	4	
2	5	5	✓
3	6	7	
4	4	9	
5	8	8	✓
6	10	10	✓
7	10	10	✓
8	12	12	✓
9	14	14	✓
10	10	13	
11	15	15	✓
12	17	17	✓
13	18	18	✓

Bảng A4.4. Bảng lịch thời gian thực hiện công việc của ví dụ 4

#	Tên công việc	Công việc chọn	Công việc loại	Đỉnh đầu	Đỉnh cuối	Tg dự phòng	Thời gian	Công việc găng
1	a			0	1	3	1	
2	b			0	2	0	5	✓
3	c			0	3	1	6	
4	d			0	4	5	4	
5	e	a		1	5	3	4	
6	f	b		2	5	0	3	✓
7	g	c		3	7	1	3	
8	h	d		4	10	5	4	
9	i	e,f		5	6	0	2	✓
10	k	g	i	7	8	0	2	✓
11	m	i		6	13	2	6	
12	l	i		6	9	1	3	
13	n	k		8	9	0	2	✓
14	o	l,n		9	11	0	1	✓
15	p	h	i,g	10	11	3	2	
16	q	h	i,g	10	12	4	3	
17	r	o,p		11	12	0	2	✓
18	s	q,r		12	13	0	1	✓



Hình A4.1: Biểu đồ Gantt lịch biểu của kế hoạch lịch, ví dụ 4



Hình A4.2: Biểu đồ sử dụng nguồn lực của kế hoạch lịch, ví dụ 4

Thời gian chạy chương trình: 0.76 giây .

## B. PHỤ LỤC 2. Mã nguồn chương trình trợ giúp quản lý dự án

### 1. Phần 1. Trang chủ

```
<html lang="en">
<head>
<meta charset="utf-8">
<!-- Viewport Meta Tag -->
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>
    Mạng công việc AOA - Luận văn thạc sĩ
</title>
<!-- Bootstrap -->
<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css">
<!-- Main Style -->
<link rel="stylesheet" type="text/css" href="assets/css/main.css">
<!-- Responsive Style -->
<link rel="stylesheet" type="text/css" href="assets/css/responsive.css">
<!--Fonts-->
<link rel="stylesheet" media="screen" href="assets/fonts/font-awesome/font-
awesome.min.css">
<link rel="stylesheet" media="screen" href="assets/fonts/simple-line-icons.css">

<!-- Extras -->
<link rel="stylesheet" type="text/css" href="assets/extras/owl/owl.carousel.css">
<link rel="stylesheet" type="text/css" href="assets/extras/owl/owl.theme.css">
<link rel="stylesheet" type="text/css" href="assets/extras/animate.css">
<link rel="stylesheet" type="text/css" href="assets/extras/normalize.css">
<link rel="stylesheet" type="text/css" href="assets/extras/settings.css">

<!-- Color CSS Styles -->
<link rel="stylesheet" type="text/css" href="assets/css/colors/green.css"
media="screen" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media
queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js">
</script>
<script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js">
</script>
<![endif]-->
<style type="text/css">
    body {
        font-size: 1em;
```

```

    }
</style>
</head>
<body>

<!-- Header area wrapper starts -->
<header id="header-wrap">

<!-- Roof area starts -->

<!-- Roof area Ends -->

<!-- Header area starts -->
<section id="header">

        <jsp:include page="_menu.jsp"></jsp:include>

</section>
<!-- Main Carousel Section -->
<div id="carousel-area">
<div id="carousel-slider" class="carousel slide" data-ride="carousel">
<!-- Indicators -->
<ol class="carousel-indicators">
<li data-target="#carousel-slider" data-slide-to="0" class="active"></li>
<li data-target="#carousel-slider" data-slide-to="1"></li>
<li data-target="#carousel-slider" data-slide-to="2"></li>
</ol>
<div class="carousel-inner" role="listbox">
<div class="carousel-item active">

<div class="carousel-caption">
<h2>
        Mạng công việc AOA
</h2>
<h3>Bài toán cân đối nguồn lực bằng phương pháp lập mạng AOA</h3>
<a class="btn btn-lg btn-common" href="projectList">
<i class="fa fa-download">
</i>
        Dự án
</a>
</div>
</div>
<div class="carousel-item">

```

```


<div class="carousel-caption">
<h2>
    Cân đối nguồn lực
</h2>
<h3>
    Cân đối nguồn lực dựa vào biểu đồ cân đối nguồn lực
</h3>
</div>
</div>
<div class="carousel-item">

<div class="carousel-caption">
<h2>
    Đơn giản
</h2>
<h3>
    Sử dụng thuật toán đơn giản, hiệu quả lập mạng công việc
</h3>
</div>
</div>
</div>
</div>
<a class="left carousel-control nav-prev" href="#carousel-slider" role="button"
data-slide="prev">
<span class="sr-only">Previous</span>
</a>
<a class="right carousel-control nav-next" href="#carousel-slider" role="button"
data-slide="next">
<span class="sr-only">Next</span>
</a>
</div>
</div>
<!-- Main Carousel Section End-->
</header>
<!-- Header-wrap Section End -->

<!-- Service Block-1 Section -->
<section id="service-block-main" class="section">
<!-- Container Starts -->
<div class="container">
<div class="row">
<h1 class="section-title wow fadeIn animated" data-wow-delay=".2s">
    Mạng công việc AOA

```



```

</h1>
<p class="section-subcontent">Giải quyết bài toán lập mạng AOA và cân đối
nguồn lực sử dụng mạng AOA</p>

</div>

<c:forEach items="${projectList}" var="project" varStatus="rowCounter">
    <c:if test="${rowCounter.count % 3 == 1}">
        <div class="row">
            </c:if>
            <div class="col-sm-4" style="text-align: center;">
                <strong style="text-align: center;">
                    ${ project.tenDA } ${
project.phbanHienTai }
                </strong>
                <a href="viewJobs?projectId=${ project.maDA
}&phban=${ project.phbanHienTai }">
                    <c:choose>
                        <c:when test="${ project.lichDA != \"\"
}">
                            
                        </c:when>
                        <c:otherwise>
                            
                        </c:otherwise>
                    </c:choose>
                </a>
                <br/>
            </div>
            <c:if test="${rowCounter.count % 3 == 0||rowCounter.count ==
fn:length(values)}">
                </div>
            </c:if>
        </c:forEach>
</div><!-- Container Ends -->
</section><!-- Service Main Section Ends -->

    <jsp:include page="_footer.jsp"></jsp:include>
<!-- Go To Top Link -->

```

```

<a href="#" class="back-to-top">
<i class="fa fa-angle-up">
</i>
</a>

<div class="bottom"><a href="#" class="settings"></a></div>

<!-- JavaScript & jQuery Plugins -->
<!-- jQuery Load -->
<script src="assets/js/jquery-min.js"></script>
<!-- Bootstrap JS -->
<script src="assets/js/bootstrap.min.js"></script>
<!--Text Rotator-->
<script src="assets/js/jquery.mixitup.js"></script>
<!--WOW Scroll Spy-->
<script src="assets/js/wow.js"></script>
<!-- OWL Carousel -->
<script src="assets/js/owl.carousel.js"></script>

<!-- WayPoint -->
<script src="assets/js/waypoints.min.js"></script>
<!-- CounterUp -->
<script src="assets/js/jquery.counterup.min.js"></script>
<!-- ScrollTop -->
<script src="assets/js/scroll-top.js"></script>
<!-- Appear -->
<script src="assets/js/jquery.appear.js"></script>
<script src="assets/js/jquery.vide.js"></script>
<!-- All JS plugin Triggers -->
<script src="assets/js/main.js"></script>
<script src="assets/js/color-switcher.js"></script>

</body>
</html>

```

## 2. Phần 2. Cập nhật dự án

### 2a. Danh sách dự án

```

public class ProjectListServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public ProjectListServlet() {

```

```

    super();
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    Connection conn = MyUtils.getStoredConnection(request);

    String errorString = null;
    List<Project> list = null;
    try {
        list = DBUtils.queryProject(conn);
    } catch (SQLException e) {
        e.printStackTrace();
        errorString = e.getMessage();
    }

    // Store info in request attribute, before forward to views
    request.setAttribute("errorString", errorString);
    request.setAttribute("projectList", list);

    // Forward to /WEB-INF/views/productListView.jsp
    RequestDispatcher dispatcher = request.getServletContext()
        .getRequestDispatcher("/WEB-INF/views/projectListView.jsp");
    dispatcher.forward(request, response);
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

## ***2b. Lập một dự án mới***

```

public class DoCreateProjectServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public DoCreateProjectServlet() {

```

```

        super();
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        Connection conn = MyUtils.getStoredConnection(request);

        String maDA = (String) request.getParameter("maDA");
        String tenDA = (String) request.getParameter("tenDA");
        String nguoiQL = (String) request.getParameter("nguoiQL");
        long tongVon = Long.parseLong(request.getParameter("tongVon"));
        String ngayBDText = (String) request.getParameter("ngayBD");
        String ngayKTText = (String) request.getParameter("ngayKT");

        Date ngayBD = null;
        Date ngayKT = null;
        DateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");

        try {
            ngayKT = formatter.parse(ngayKTText);
            ngayBD = formatter.parse(ngayBDText);
        } catch (ParseException pe) {
            // TODO Auto-generated catch block
            pe.printStackTrace();
        }

        Project project = new Project();
        project.setMaDA(maDA);
        project.setTenDA(tenDA);
        project.setNguoiQL(nguoiQL);
        project.setTongVon(tongVon);
        project.setNgayBD(ngayBD);
        project.setNgayKT(ngayKT);
        String errorString = null;

        if (errorString == null) {
            try {
                DBUtils.insertProject(conn, project);
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}

```

```

        errorString = e.getMessage();
    }
}

// Store information to request attribute, before forward to views.
request.setAttribute("errorString", errorString);
request.setAttribute("project", project);

// If error, forward to Edit page.
if (errorString != null) {
    RequestDispatcher dispatcher = request.getRequestDispatcher()
        .getRequestDispatcher("/WEB-INF/views/createProjectView.jsp");
    dispatcher.forward(request, response);
}

// If everything nice.
// Redirect to the product listing page.
else {
    response.sendRedirect(request.getContextPath() + "/projectList");
}
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

### 2c. Tạo một dự án mới

```

</h1>
<p class="section-subcontent">Tạo một dự án mới để tiến hành thử nghiệm lập
mạng AOA</p>
<div class="col-lg-12">
    <form method="POST"
action="doCreateProject">
        <div class="form-group">
            <label for="maDA">Mã dự án:</label>

```

```

control" id="maDA">
    <input name="maDA" type="text" class="form-
    </div>
    <div class="form-group">
    <label for="tenDA">Tên dự án:</label>
control" id="tenDA">
    <input name="tenDA" type="text" class="form-
    </div>
    <div class="form-group">
    <label for="nguoiquanly">Người quản lý:</label>
name="nguoiquanly"/>
    <input class="form-control" id="nguoiquanly"
    </div>
    <div class="form-group">
    <label for="tongvon">Tổng vốn:</label>
name="tongvon"/>
    <input class="form-control" id="tongvon"
    </div>
    <div class="form-group">
    <label for="ngaybd">Ngày bắt đầu:</label>
    <div id="ngaybdpicker" class="input-append">
        <input data-format="dd/MM/yyyy"
class="form-control" id="ngaybd" name="ngaybd" />
        <span class="add-on">
            <i data-time-icon="icon-time" data-
date-icon="icon-calendar">
                </i>
            </span>
        </div>
    </div>
    <div class="form-group">
    <label for="ngaykt">Ngày kết thúc:</label>
    <div id="ngayktpicker" class="input-append">
        <input data-format="dd/MM/yyyy"
class="form-control" id="ngaykt" name="ngaykt" />
        <span class="add-on">
            <i data-time-icon="icon-time" data-
date-icon="icon-calendar">
                </i>
            </span>
        </div>
    </div>
    </div>
    </div>

```

```

                <button type="submit" class="btn btn-
success"><i class="fa fa-plus-square"></i> Tạo dự án</button>
                </form>
</div>

</div>
</div><!-- Container Ends -->
</section>
<!-- Services Section End -->

<!-- Footer Section -->
<footer>
<!-- Container Starts -->
<div class="container">
<!-- Row Starts -->
<div class="row section">
<!-- Footer Widget Starts -->
<div class="footer-widget col-md-3 col-xs-12 wow fadeIn">
<h3 class="small-title">
    Về tôi Trần Nhật Giáp
</h3>
<div class="social-footer">
<a href="#"><i class="fa fa-facebook icon-round"></i></a>
<a href="#"><i class="fa fa-twitter icon-round"></i></a>
<a href="#"><i class="fa fa-linkedin icon-round"></i></a>
<a href="#"><i class="fa fa-google-plus icon-round"></i></a>
</div>
</div><!-- Footer Widget Ends -->
</div><!-- Row Ends -->
</div><!-- Container Ends -->

<!-- Copyright -->
<div id="copyright">
<div class="container">
<div class="row">
<div class="col-md-6 col-sm-6">
<p class="copyright-text">
    © 2016 tất cả các quyền đã được bảo hộ. Được thiết kế bởi <a
href="#">Trần Nhật Giáp</a>
</p>
</div>
<div class="col-md-6 col-sm-6">

```

```

<ul class="nav nav-inline pull-xs-right">
<li class="nav-item">
<a class="nav-link active" href="home">Trang chủ</a>
</li>
<li class="nav-item">
<a class="nav-link" href="#">Sitemap</a>
</li>
<li class="nav-item">
<a class="nav-link" href="#">Chính sách</a>
</li>
<li class="nav-item">
<a class="nav-link" href="#">Dịch vụ</a>
</li>
</ul>
</div>
</div>
</div>
</div>
<!-- Copyright End-->

</footer>
<!-- Footer Section End-->
<!-- Go To Top Link -->
<a href="#" class="back-to-top">
<i class="fa fa-angle-up">
</i>
</a>

<div class="bottom"><a href="#" class="settings"></a></div>

<!-- JavaScript & jQuery Plugins -->
<!-- jQuery Load -->
<script src="assets/js/jquery-min.js"></script>
<!-- Bootstrap JS -->
<script src="assets/js/bootstrap.min.js"></script>
<!--Text Rotator-->
<script src="assets/js/jquery.mixitup.js"></script>
<!--WOW Scroll Spy-->
<script src="assets/js/wow.js"></script>
<!-- OWL Carousel -->
<script src="assets/js/owl.carousel.js"></script>

<!-- WayPoint -->

```



```

<script src="assets/js/waypoints.min.js"></script>
<!-- CounterUp -->
<script src="assets/js/jquery.counterup.min.js"></script>
<!-- ScrollTop -->
<script src="assets/js/scroll-top.js"></script>
<!-- Appear -->
<script src="assets/js/jquery.appear.js"></script>
<script src="assets/js/jquery.vide.js"></script>
<!-- All JS plugin Triggers -->
<script src="assets/js/main.js"></script>
<script src="assets/js/color-switcher.js"></script>

<script src="js/bootstrap-datetimepicker.min.js"></script>

<script type="text/javascript">
$(function() {
    $('#ngaybdpicker').datetimepicker({
        pickTime: false
    });
    $('#ngayktpicker').datetimepicker({
        pickTime: false
    });
});
</script>

</body>
</html>

```

## ***2d. Lập lịch Dự án mới***

```

public class DoCreateProjectServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public DoCreateProjectServlet() {
        super();
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        Connection conn = MyUtils.getStoredConnection(request);

        String maDA = (String) request.getParameter("maDA");

```

```

String tenDA = (String) request.getParameter("tenDA");
String nguoiQL = (String) request.getParameter("nguoiQL");
long tongVon = Long.parseLong(request.getParameter("tongVon"));
String ngayBDText = (String) request.getParameter("ngayBD");
String ngayKTText = (String) request.getParameter("ngayKT");

Date ngayBD = null;
Date ngayKT = null;
DateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");

try {
    ngayKT = formatter.parse(ngayKTText);
    ngayBD = formatter.parse(ngayBDText);
} catch (ParseException pe) {
    // TODO Auto-generated catch block
    pe.printStackTrace();
}

Project project = new Project();
project.setMaDA(maDA);
project.setTenDA(tenDA);
project.setNguoiQL(nguoiQL);
project.setTongVon(tongVon);
project.setNgayBD(ngayBD);
project.setNgayKT(ngayKT);
String errorString = null;

if (errorString == null) {
    try {
        DBUtils.insertProject(conn, project);
    } catch (SQLException e) {
        e.printStackTrace();
        errorString = e.getMessage();
    }
}

// Store information to request attribute, before forward to views.
request.setAttribute("errorString", errorString);
request.setAttribute("project", project);

// If error, forward to Edit page.
if (errorString != null) {

```

```

        RequestDispatcher dispatcher = request.getServletContext()
            .getRequestDispatcher("/WEB-INF/views/createProjectView.jsp");
        dispatcher.forward(request, response);
    }

    // If everything nice.
    // Redirect to the product listing page.
    else {
        response.sendRedirect(request.getContextPath() + "/projectList");
    }

}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

### 3. Phần 3. Lập lịch cho dự án

```

public class RunAOAServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public RunAOAServlet() {
        super();
    }

    boolean EQUALS(LinkedList<String> a, LinkedList<String> b) {
        for (int i = 0; i < a.size(); i++) {
            if (!a.get(i).equals(b.get(i))) {
                return false;
            }
        }
        return true;
    }

    boolean CONTAINSALL(LinkedList<String> a, LinkedList<String> b) {
        for (int i = 0; i < b.size(); i++) {

```

```

        boolean flag = false;
        for (int j = 0; j < a.size(); j++) {
            if (b.get(i).equals(a.get(j))) {
                flag = true;
                break;
            }
        }
        if (!flag) return false;
    }

    return true;
}

int LEFMID(LinkedList<String> CVchi, int j, String[] CV) {
    for (int i = 0; i < CV.length; i++) {
        if (CVchi.get(j).equals(CV[i])) {
            return i;
        }
    }
    return -1;
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
    Connection conn = MyUtils.getStoredConnection(request);

    String maDA = request.getParameter("id");
    String phbanDA = request.getParameter("phban");
    String errorString = null;

    List<Job> list = null;
    try {
        list = DBUtils.queryJob(conn, maDA, phbanDA);
    } catch (SQLException e) {
        e.printStackTrace();
        errorString = e.getMessage();
    }

    String[] CV = new String[list.size()];
    int i = 0;
    int N = list.size();

```

```

long startTime = System.nanoTime();

int[] tgCV = new int[N];
    int[] nlCV = new int[N];
    int[] thutuCV = new int[N];

    List<LinkedList<String>> CVtr = new
ArrayList<LinkedList<String>>();
    for (i = 0; i < N; i++) {
        CVtr.add(new LinkedList<String>());
    }
    List<LinkedList<String>> CVtr2 = new
ArrayList<LinkedList<String>>(CVtr.size());

    List<LinkedList<String>> CVch = new
ArrayList<LinkedList<String>>();
    for (i = 0; i < N; i++) {
        CVch.add(new LinkedList<String>());
    }
    List<LinkedList<String>> CVlo = new
ArrayList<LinkedList<String>>();
    for (i = 0; i < N; i++) {
        CVlo.add(new LinkedList<String>());
    }
    List<LinkedList<String>> CVxoa = new
ArrayList<LinkedList<String>>();
    for (i = 0; i < N; i++) {
        CVxoa.add(new LinkedList<String>());
    }

    i = 0;

    for (Job j : list) {
        CV[i] = j.getTenCV();
        tgCV[i] = j.getThoigianCV();
        nlCV[i] = j.getNguonlucCV();
        thutuCV[i] = j.getThutuCV();

        if (j.getCvDiTruoc() != null) {
            String[] pjs = j.getCvDiTruoc().split(",");
            for (String tmp : pjs) {
                CVtr.get(j.getThutuCV() - 1).add(tmp);
            }
        }
    }

```

```

    }
    }
    i++;
}

```

```

for (i = 0; i < CVtr.size(); i++) {
    CVtr2.add((LinkedList<String>)CVtr.get(i).clone());
}

```

```

boolean flag = false;

```

```

while (!flag) {
    flag = true;
    int k = N;
    int smin = Integer.MAX_VALUE;
    for (i = 0; i < k; i++) {
        int l = CVtr2.get(i).size();
        if (l == 0) {
            continue;
        }
        if (l == 1) {
            smin = 1;
            break;
        }
        if (l < smin) {
            smin = l;
        }
    }
}

```

```

int j = 0;
for (i = 0; i < k; i++) {
    int l = CVtr2.get(i).size();
    if (l == smin) {
        CVch.get(i).addAll(CVtr2.get(i));
        CVxoa.get(j).clear();
        CVxoa.get(j).addAll(CVtr2.get(i));
        CVtr2.get(i).clear();
        j++;
    }
}
int J = j;

```

```

for (i = 0; i < k - 1; i++) {

```

```

        for (int v = i + 1; v < CVxoa.size(); v++) {
            if (CVxoa.get(i).size() > 0 &&
CVxoa.get(v).size() > 0 && CVxoa.get(i).equals(CVxoa.get(v))) {
                CVxoa.remove(v);
                J--;
            }
        }
    }

    for (i = 0; i < k; i++) {
        if (CVtr2.get(i).size() > smin) {
            for (int v = 0; v < J; v++) {
                if (CVxoa.get(v).size() > 0) {
                    if
(CONTAINSALL(CVtr2.get(i), CVxoa.get(v))) {
                        //if
(CVtr2.get(i).containsAll(CVxoa.get(v))) {

                            CVtr2.get(i).removeAll(CVxoa.get(v));

                            CVlo.get(i).addAll(CVxoa.get(v));
                        }
                    }
                }
            }
        }
        for (i = 0; i < k; i++) {
            if (CVtr2.get(i).size() > 0) {
                flag = false;
                break;
            }
        }
    }

    int[] ddCV = new int[N * 2];
    int[] dcCV = new int[N * 2];

    int[] X = new int [N];
    for (i = 0; i < N; i++) {
        X[i] = 1;
    }
    for (i = 0; i < N; i++) {
        if (CVtr.get(i).size() == 0) {

```

```

        ddCV[i] = 0;
        X[i] = 0;
    }
}

int k = 1;
for (i = 0; i < N; i++) {
    if (X[i] == 1) {
        if (CVch.get(i).size() == 0) {
            ddCV[i] = k;
            X[i] = 0;
            k++;
        }
        else {
            boolean flagTuple = false;

            int vI = 0;
            for (int v = i - 1; v >= 0; v--) {
                if (CVch.get(v).size() > 0 &&
CVch.get(i).size() > 0 && EQUALS(CVch.get(v), CVch.get(i))) {
                    flagTuple = true;
                    vI = v;
                    break;
                }
            }
            if (!flagTuple) {
                int L = CVch.get(i).size();
                int j = 0;
                int m = L;
                for (j = 0; j < L; j++) {
                    int l =
LEFMID(CVch.get(i), j, CV);

                    if (l >= 0) {
                        X[l] = 0;
                        m--;
                    }
                }
                if (m == 0) {
                    for (j = 0; j < L; j++) {
                        int l =
LEFMID(CVch.get(i), j, CV);

                        dcCV[l] = k;
                    }
                }
            }
        }
    }
}

```



```

ddCV[i] = k;
X[i] = 0;
k++;
    }
    }
else {
    ddCV[i] = ddCV[vI];
}
}
}
}

for (i = 0; i < N; i++) {
    if (dcCV[i] == 0) {
        dcCV[i] = k;
    }
}

i = 0;
int M = N;
while (i < N) {
    if (CVlo.get(i).size() > 0) {
        int L = CVlo.get(i).size();
        int j = 0;
        for (j = 0; j < L; j++) {
            int k1 = LEFMID(CVlo.get(i), j, CV);
            ddCV[M + j] = dcCV[k1];
            dcCV[M + j] = ddCV[i];
        }
        M = M + L;
    }
    i++;
}

while (i < M) {
    if (ddCV[i] > dcCV[i]) {
        int k1 = ddCV[i];
        int d1 = dcCV[i];
        for (int j = 0; j < M; j++) {
            if (j != i) {
                if (ddCV[j] == k1) {
                    ddCV[j] = d1;
                }
                else if (ddCV[j] == d1) {

```

```

        ddCV[j] = k1;
    }
    else if (dcCV[j] == k1) {
        dcCV[j] = d1;
    }
    else if (dcCV[j] == d1) {
        dcCV[j] = k1;
    }
}
}
ddCV[i] = d1;
dcCV[i] = k1;
i = 0;
}
i++;
}

int[] ts = new int[N];
ts[0] = 0;
for (i = 1; i <= k; i++) {
    int max = 0;
    for (int j = 0; j < M; j++) {
        if (dcCV[j] == i) {
            if (j < N) {
                if (max < (ts[ddCV[j]] +
                    tgCV[j])) {
                    max = ts[ddCV[j]] +
                    tgCV[j];
                }
            }
            else if (max < ts[ddCV[j]]) {
                max = ts[ddCV[j]];
            }
        }
    }

    ts[i] = max;
}

int[] tm = new int[N];
tm[k] = ts[k];
for (i = k - 1; i >= 0; i--) {
    int min = Integer.MAX_VALUE;

```

```

        for (int j = 0; j < M; j++) {
            if (ddCV[j] == i) {
                if (j < N) {
                    if (min > (tm[dcCV[j]] -
tgCV[j])) {
                        min = tm[dcCV[j]] -
tgCV[j];
                    }
                }
            }
            else if (min > tm[dcCV[j]]) {
                min = tm[dcCV[j]];
            }
        }
        tm[i] = min;
    }

    int[] tdf = new int[N];
    for (i = 0; i < N; i++) {
        tdf[i] = tm[dcCV[i]] - ts[ddCV[i]] - tgCV[i];
    }

// Store info in request attribute, before forward to views
request.setAttribute("errorString", errorString);
for (i = 0; i < N; i++) {
    StringBuilder tmp = new StringBuilder();
    for (String s : CVch.get(i)) {
        tmp.append(s);
        tmp.append(",");
    }
    if (tmp.length() > 0) {
        tmp.deleteCharAt(tmp.length() - 1);
    }

    list.get(i).setCvCh(tmp.toString());
}
for (i = 0; i < N; i++) {
    StringBuilder tmp = new StringBuilder();
    for (String s : CVlo.get(i)) {
        tmp.append(s);
        tmp.append(",");
    }
}

```

```

        if (tmp.length() > 0) {
            tmp.deleteCharAt(tmp.length() - 1);
        }

        list.get(i).setCvLo(tmp.toString());
    }

    for (i = 0; i < N; i++) {
        if (tdf[i] == 0) {
            list.get(i).setGantt(true);
        }
        list.get(i).setDdcv(ddCV[i]);
        list.get(i).setDccv(dcCV[i]);
        list.get(i).setTdf(tdf[i]);
        list.get(i).setTs(ts[ddCV[i]]);
        list.get(i).setTm(tm[dcCV[i]]);
    }

    List<AOANetwork> aoaList = new ArrayList<>(k);
    for (i = 0; i <= k; i++) {
        AOANetwork aoa = new AOANetwork();
        aoa.setTs(ts[i]);
        aoa.setTm(tm[i]);
        aoa.setTdf(tdf[i]);
        aoa.setVertice(i);
        aoa.setTenCV(CV[i]);
        aoa.setThutuCV(thutuCV[i]);
        aoa.setDinhDau(ddCV[i]);
        aoa.setDinhCuoi(dcCV[i]);
        aoa.setThoigianCV(tgCV[i]);
        aoa.setThoigianBDCV(ts[i]);
        aoa.setThoigianKTCV(tm[i]);
        aoa.setThoigianDFCV(tdf[i]);
        aoa.setNguonlucCV(nlCV[i]);

        if (tm[i] == ts[i]) {
            aoa.setGantt(true);
        }
        else {
            aoa.setGantt(false);
        }
        aoaList.add(aoa);
    }

```

```

}
int maxTime = 0;
for (i = 0; i < N; i++) {
    if (tm[i] > maxTime) maxTime = tm[i];
}

long endTime = System.nanoTime();
long totalTime = endTime - startTime;

int[] tgNI = new int[maxTime + 1];
tgNI[0] = 0;
int maxNI = 0;

for (i = 1; i <= maxTime; i++) {
    tgNI[i] = 0;
    for (int j = 0; j < N; j++) {
        if ((ts[ddCV[j]] + 1) <= i && (i <= ts[ddCV[j]] + tgCV[j])) {
            tgNI[i] += nlCV[j];
        }
    }
    if (maxNI < tgNI[i]) {
        maxNI = tgNI[i];
    }
}

List<Chart> chartList = new ArrayList<>();
Chart cp = new Chart();
cp.setX(0);
cp.setY(0);
chartList.add(cp);
int prevY = 0;
for (i = 1; i < tgNI.length; i++) {
    if (prevY != tgNI[i]) {
        cp = new Chart();
        cp.setX(i);
        cp.setY(prevY);
        chartList.add(cp);
    }

    cp = new Chart();
    cp.setX(i);
    cp.setY(tgNI[i]);

    chartList.add(cp);
}

```

```

        prevY = tgNI[i];
    }

    cp = new Chart();
    cp.setX(maxTime);
    cp.setY(0);

    chartList.add(cp);

    Date now = new Date();

    try {
        DBUtils.deleteProjectVersionByProject(conn, maDA,
phbanDA);
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    for (Job j : list) {
        ProjectNetwork pn = new ProjectNetwork();
        pn.setMaDA(maDA);
        pn.setPhbanDA(phbanDA);
        pn.setNgaySua(new java.sql.Date(now.getTime()));
        pn.setCvGang(j.isGantt() ? 1 : 0);
        pn.setThutuCV(j.getThutuCV());
        pn.setTenCV(j.getTenCV());
        pn.setDinhDau(j.getDdcv());
        pn.setDinhCuoi(j.getDccv());
        pn.setThoigianCV(j.getThoigianCV());
        pn.setThoigianBDCV(ts[j.getDdcv()]);
        pn.setThoigianKTCV(tm[j.getDccv()]);
        pn.setThoigianDFCV(j.getTdf());
        pn.setNguonlucCV(j.getNguonlucCV());

        try {
            DBUtils.insertProjectNetwork(conn, pn);
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}

```

```

request.setAttribute("jobList", list);
request.setAttribute("chartList", chartList);
request.setAttribute("aoaList", aoaList);
request.setAttribute("maxTime", maxTime);
request.setAttribute("maxNI", maxNI);
NumberFormat formatter = new DecimalFormat("#0.00000");
request.setAttribute("runningTime", formatter.format(totalTime * 1.0 /
1000000000));
request.setAttribute("maDA", maDA);
request.setAttribute("phbanDA", phbanDA);

// Forward to /WEB-INF/views/productListView.jsp
RequestDispatcher dispatcher = request.getRequestDispatcher()
    .getRequestDispatcher("/WEB-INF/views/runAOAView.jsp");
dispatcher.forward(request, response);
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        doGet(request, response);
    }
}

```

## 4. Phần 4. Hệ thống

### 4a. Thông tin người dùng

```

public class UserListServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public UserListServlet() {
        super();
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        Connection conn = MyUtils.getStoredConnection(request);

        String errorString = null;
        List<UserAccount> list = null;
        try {
            list = DBUtils.queryUserAccount(conn);

```

```

    } catch (SQLException e) {
        e.printStackTrace();
        errorString = e.getMessage();
    }

    // Store info in request attribute, before forward to views
    request.setAttribute("errorString", errorString);
    request.setAttribute("userList", list);

    // Forward to /WEB-INF/views/userListView.jsp
    RequestDispatcher dispatcher = request.getServletContext()
        .getRequestDispatcher("/WEB-INF/views/userListView.jsp");
    dispatcher.forward(request, response);
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

#### 4.b. Thông tin cá nhân

```

public class DoEditUserInfoServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public DoEditUserInfoServlet() {
        super();
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        Connection conn = MyUtils.getStoredConnection(request);

        String maNgDung = request.getParameter("maNgDung");
        String tenNgDung = request.getParameter("tenNgDung");
        String ngaySinhNDText = (String)
request.getParameter("ngaySinhND");

```



```

DateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");
Date ngaySinhND = null;
try {
    ngaySinhND = formatter.parse(ngaySinhNDText);
} catch (ParseException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
String mailND = (String) request.getParameter("mailND");
String teleND = (String) request.getParameter("teleND");
String passwordND = (String)
request.getParameter("passwordND");
HttpSession session = request.getSession();

UserAccount user = MyUtils.getLoginedUser(session);

user.setMaNgDung(maNgDung);
user.setTenNgDung(tenNgDung);
user.setNgaySinhND(ngaySinhND);
user.setMailND(mailND);
user.setTeleND(teleND);
if (passwordND != null && !passwordND.equals("")) {

user.setPasswordND(DigestUtils.md5Hex(passwordND));
}

String errorString = null;

if (errorString == null) {
    try {
        DBUtils.updateUserAccount(conn, user);
        MyUtils.storeLoginedUser(session, user);
    } catch (SQLException e) {
        e.printStackTrace();
        errorString = e.getMessage();
    }
}

// Store infomation to request attribute, before forward to views.
request.setAttribute("errorString", errorString);
request.setAttribute("userAccount", user);

// If error, forward to Edit page.

```

```

        if (errorString != null) {
            RequestDispatcher dispatcher =
request.getServletContext()
                .getRequestDispatcher("/WEB-
INF/views/editUserInfoView.jsp");
            dispatcher.forward(request, response);
        }

        // If everything nice.
        // Redirect to the product listing page.
        else {
            response.sendRedirect(request.getContextPath() +
"/userInfo");
        }
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        doGet(request, response);
    }
}

```

## 5. Phần 5. Đăng nhập và đăng xuất

### a. Đăng nhập

```

public class DoLoginServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    public DoLoginServlet() {
        super();
    }

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        String mailND = request.getParameter("mailND");
        String passwordND = request.getParameter("passwordND");
        String rememberMeStr = request.getParameter("rememberMe");
        boolean remember= "Y".equals(rememberMeStr);
    }
}

```

```

UserAccount user = null;
boolean hasError = false;
String errorString = null;

if (mailND == null || passwordND == null
    || mailND.length() == 0 || passwordND.length() == 0) {
    hasError = true;
    errorString = "Bạn phải nhập vào tài khoản và mật khẩu!";
} else {
    Connection conn = MyUtils.getStoredConnection(request);
    try {

        user = DBUtils.findUser(conn, mailND,
DigestUtils.md5Hex(passwordND));

        if (user == null) {
            hasError = true;
            errorString = "Tài khoản hoặc mật khẩu không đúng";
        }
        else if (user.getKichHoat() == 0) {
            hasError = true;
            errorString = "Tài khoản hoặc mật khẩu không đúng";
        }
    } catch (SQLException e) {
        e.printStackTrace();
        hasError = true;
        errorString = e.getMessage();
    }
}

// If error, forward to /WEB-INF/views/loginView.jsp
if (hasError) {
    user = new UserAccount();
    user.setMailND(mailND);
    user.setPasswordND(passwordND);

    // Store information in request attribute, before forward.
    request.setAttribute("errorString", errorString);
    request.setAttribute("user", user);
}

```

```

        // Forward to /WEB-INF/views/login.jsp
        RequestDispatcher dispatcher //
        = this.getServletContext().getRequestDispatcher("/WEB-
INF/views/loginView.jsp");

        dispatcher.forward(request, response);
    }

    // If no error
    // Store user information in Session
    // And redirect to userInfo page.
    else {
        HttpSession session = request.getSession();
        MyUtils.storeLoggedInUser(session, user);

        // If user checked "Remember me".
        if(remember) {
            MyUtils.storeUserCookie(response, user);
        }

        // Else delete cookie.
        else {
            MyUtils.deleteUserCookie(response);
        }

        // Redirect to userInfo page.
        response.sendRedirect(request.getContextPath() + "/home");
    }
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}

```

### ***b. Đăng xuất***

```

public class DoLogoutServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

```

```
public DoLogoutServlet() {
    super();
}

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    HttpSession session = request.getSession();
    session.invalidate();
    MyUtils.deleteUserCookie(response);
    response.sendRedirect(request.getContextPath() + "/login");
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    doGet(request, response);
}
}
```