Tutorial Letter 501/0/2023

Software Project Management

INF3708

Year Module(s)

DEPARTMENT OF INFORMATION SYSTEMS

IMPORTANT INFORMATION

Please register on myUnisa, activate your myLife e-mail account and make sure that you have regular access to the myUnisa module website, INF3708-23-Y, as well as your group website.

Note: This is a fully online module. It is, therefore, only available on myUnisa.

BARCODE

Define tomorrow.

UNISA university of south africa

CONTENTS

Page

1. Introduction

Greetings fellow students! This is tutorial letter 501. In this tutorial letter, I discuss the following:

- I demonstrate how to create an activity-on-arrow (AoA) diagram.
- I discuss question 1.6.
- I discuss question 1.3.

This entry into the INF3708 tutorial letter series is inspired by the teachings of Schwalbe (2019).

2. Activity-on-arrow diagram

An activity-on-arrow and activity-on-node (AoN) are diagramming tools for scheduling and mapping the activities and tasks of a project. There are, however, differences between the two diagrams. An activity-on-arrow diagram is an appropriate tool to illustrate finish-to-start relationships, that is, mandatory dependencies. A finish-to-start relationship means some activities must be completed before another activity can start. Consider the relationship between the design and development phases of an information system; the designers must first finalise the requirements, architecture and graphical user interface before developers can begin coding. Examine Figure 1, the legend system of an AoA sequence flow. The circle represents an EVENT. For example, the first event can represent user interface design and the second event can represent user interface coding. Both events must start and finish on a specific day.

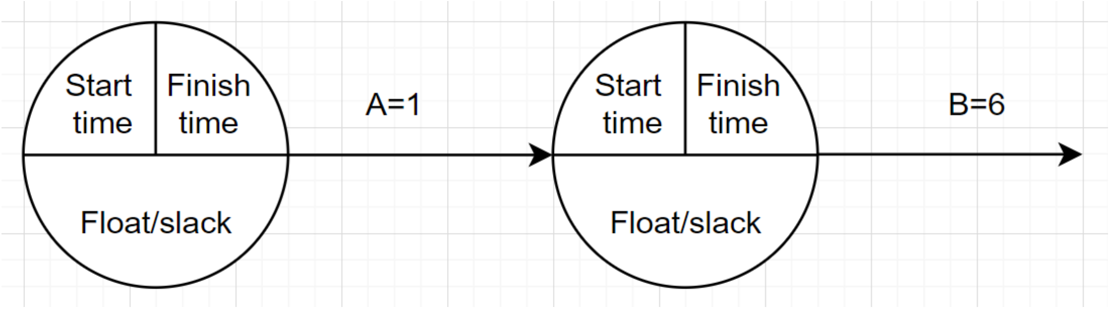

Figure 1. A legend system of an AoA sequence flow (adapted from Gilda, 2018)

Now examine Figure 2, the legend system of an AoN sequence flow. Both Activities A and B can start with a time lag with reference to early start and early finish times of an activity node.

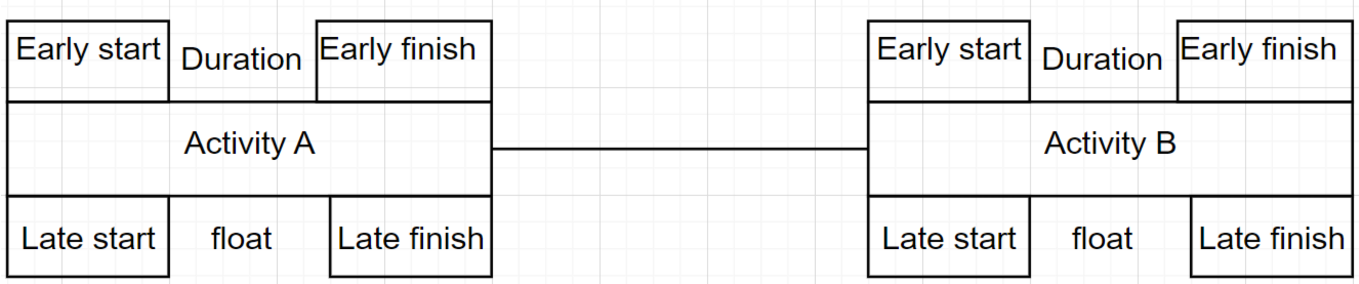

Figure 2. A legend system of an AoN sequence flow (adapted from Oguz, 2022)

2.1. Rational

For this demonstration, I adopt the AoA legend system illustrated in Figure 3.

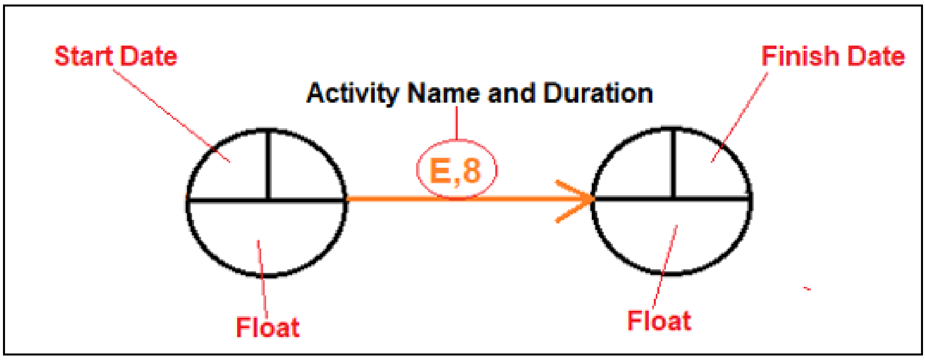

Figure 3. Activity-on-arrow legend system (Gilda, 2018)

The AoA demonstrated in this section is based on the activities listed in Table 1. The activities are embedded in a scrum project management methodology. The scrum methodology is characterised by a small team, typically software developers, who do rapid coding of an information system in short iterative cycles called sprints. Sprints are short, usually one to two weeks. In this vein, I regard the activity-on-arrow diagram as a fitting diagramming tool to schedule coding sprints. Conversely, I recommend AoN diagramming as an appropriate tool for project activities are in pending states; for example, physical installation of the tablets at Grilloo cannot start until delivery of the devices. Delivery can be delayed for various reasons, e.g., union strikes, faulty devices, supplier issues, and so forth. Nonetheless, the immediately following passages demonstrate how to create an activity-on-arrow diagram.

Table 1: Activities with associated estimated duration and predecessors.

Activity label	Activity description	Estimated duration	Predecessor
A	Programming Lead presents overview of software deliverables (object, schedule, cost)	1	-

B	Architect design (interface linkages, dependencies, database tables)	1	A
C	Modular design (create modules that can be independently modified, replaced, or exchanged with other modules between different systems)	2	A
D	Unit testing (test that smallest module can perform independent from other modules)	4	B
E	Integration testing (Module groups created independently can cohabit and interact with other modules)	5	C
F	System testing (test if the application developer’s expectations are satisfied)	1	D, E
G	Acceptance testing (test software product in real-world with end-users)	2	F

2.2. Forward pass

First draw a start node. The start node represents start of the project (or sub-project) and starts with zero (on day zero). In a detailed activity-on-arrow diagram, you will of course indicate the specific date that the project starts, for example, 31 July 2023 (so, 31 July 2023 is day zero). See Figure 4.

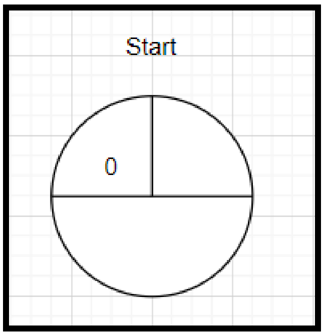

Figure 4. The start node with a start time of 0.

To determine the start time of the second event, you add the project start time of 0 to the duration of Activity A.

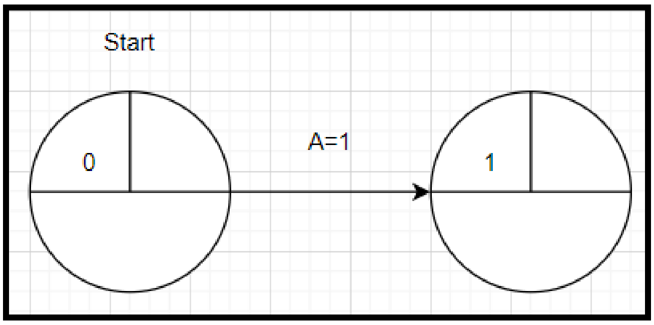

Figure 5. The start time of the second event was calculated by adding the start time of the previous event to the activity duration.

To determine the start time of each event, you add the start time of the previous event to the activity that precedes the event in question. This technique is called the **forward pass calculation**. See Figure 6.

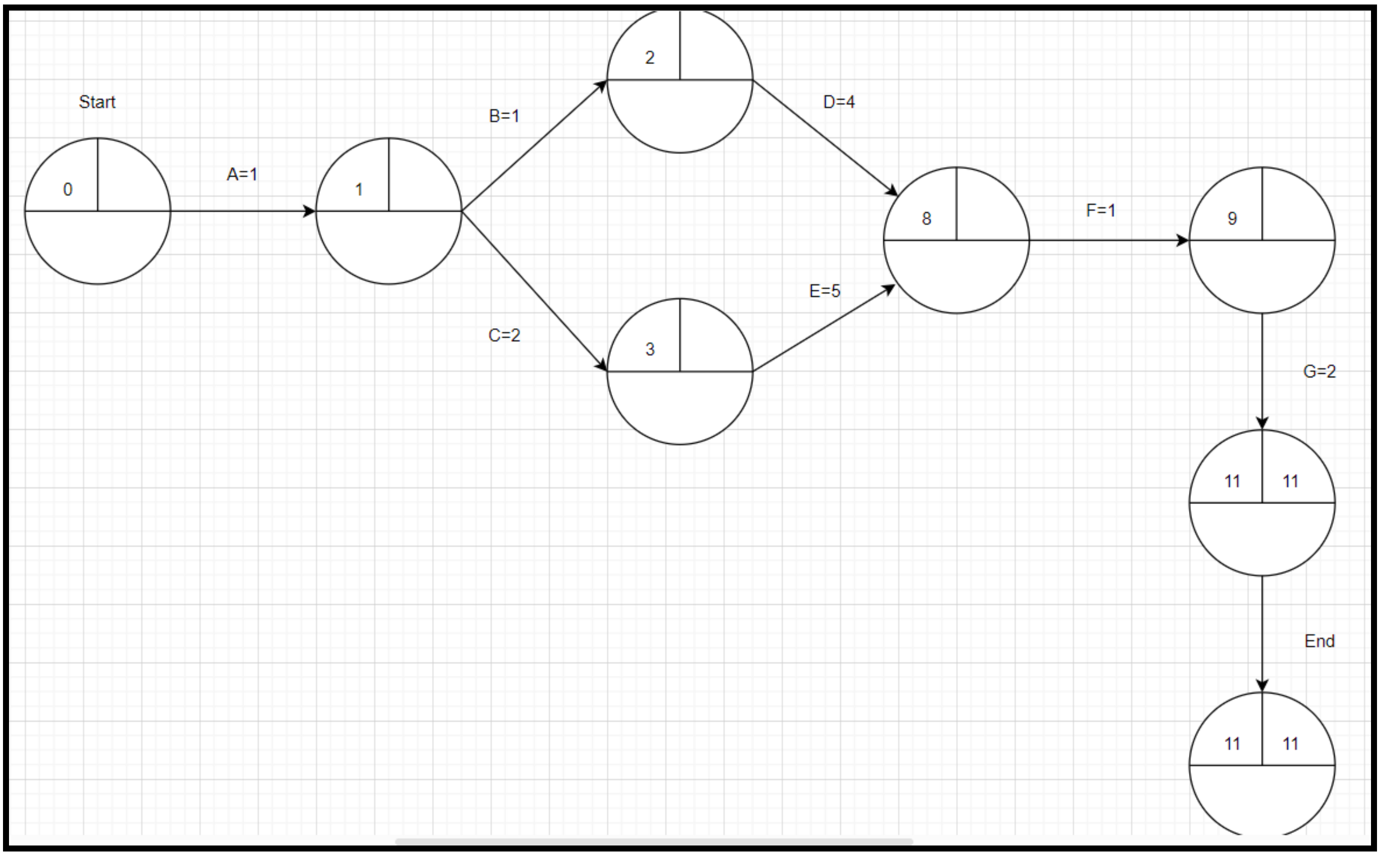

Figure 6. The start time of each event was calculated using the forward pass technique.

Figure 6 holds a few observations. In the instance where Event F is preceded by Events D and E, the latest (highest, and not earliest/lowest) start time is considered for Event F. Event F cannot start on day 6 (Event B start time of 2 + Activity D duration of 4) because Activity E must first be completed (Event C start time of 3 + Activity E duration of 5). Also, the start time of EVENT G is transferred to its end time position to indicate that the project has reached its completion. Also, an end node has been added to signify that EVENT G has no successor events.

2.3. Backward pass

To calculate the finish time of each event, you subtract its activity duration from the finish time of the successor node. This is known as the **backward pass calculaton**. See Figure 7. Additional observation can be made about Figure 7. In an instance where two or multiple events is preceded by one event (e.g., Events B and C is preceded by Event A), the lowest finish time is transferred to the predecessor. This is done to determine the earliest time at which the predecessor activity can finish without causing any delays to its successor activities.

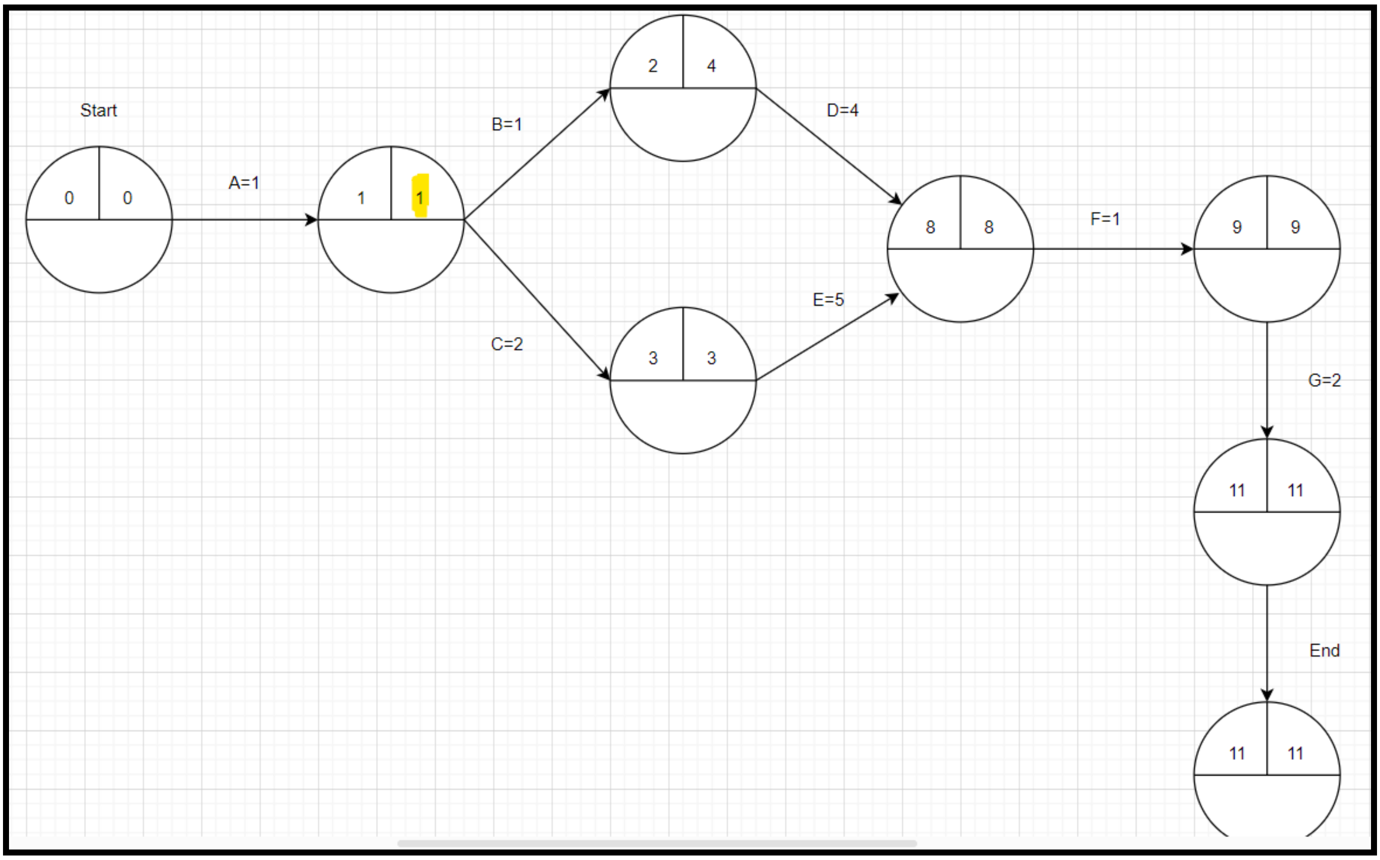

Figure 7. The finish time of each event was calculated using the backward pass technique.

After the start and finish times are calculated, you must determine the critical path and the events that indicate a slack time. The critical path is referred to as the sequence of activities that will take the longest to complete. Activities are considered critical because if a delay in its completion time will delay the completion time of the entire project. To determine the critical path, you need to first determine the float or slack. Float is the amount of time that an activity can be delayed without delaying the entire project completion date. To calculate the float of an activity, you subtract the start time of an activity from its finish date. See Figure 8.

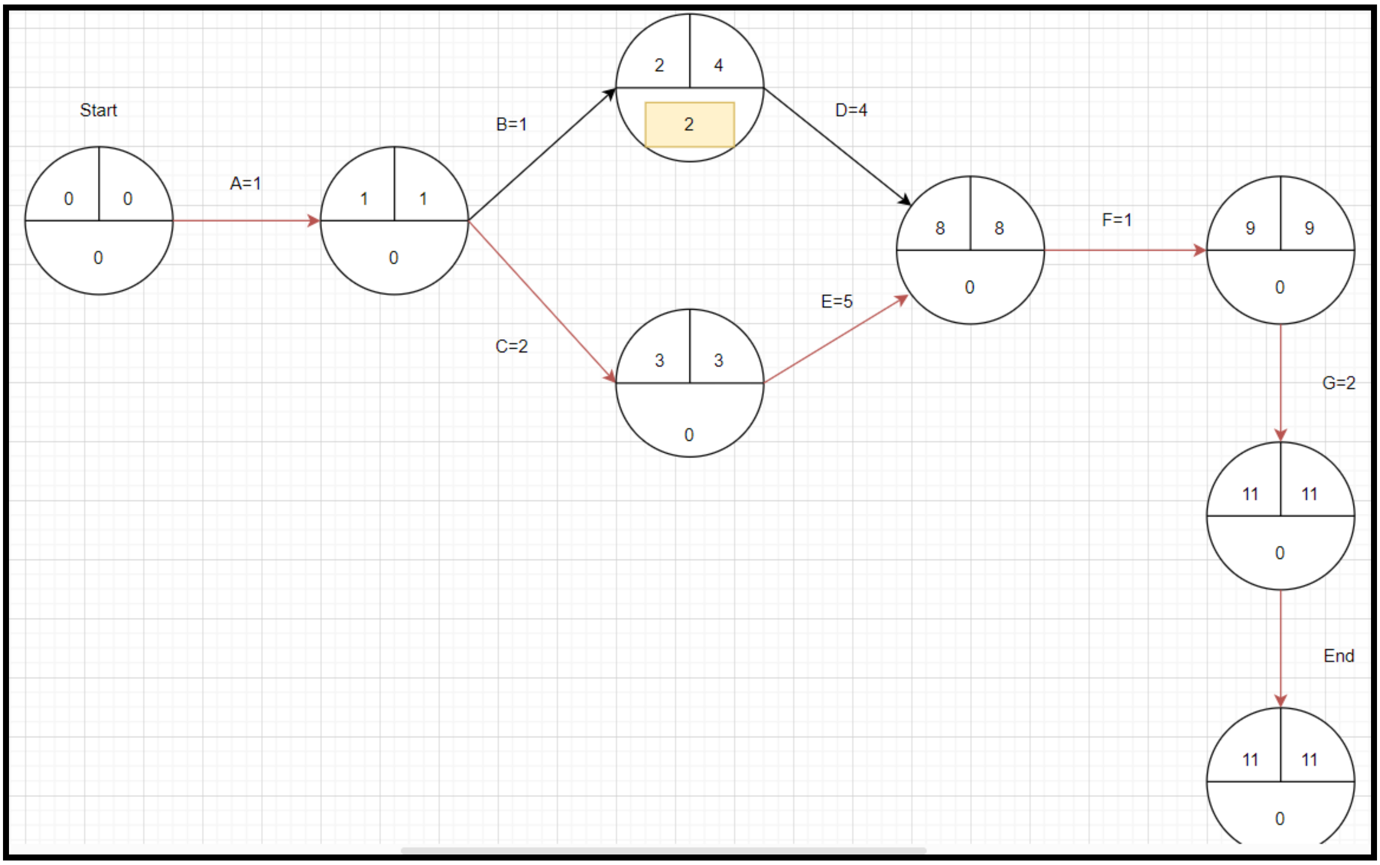

Figure 8. Activity B indicates a slack of 2 which means the critical path is A-C-E-F-G.

3. Cash inflow

I received many e-mails from fellow students, expressing concern about question 1.6. A fellow student wants to know if "[there' is sufficient info in the question … am [I] missing it[?]" Many of you insist on a presentation of incremental (or decremental) annual cash inflow and outflow. However, it is possible to make a once-off investment at the start of the project (Year 0) and not incur further expenses again. Consider the physical and software installation of a solar panel system. The initial installation is expensive, but the money you save on electricity will pay for the solar panel system in seven years or less.

Also keep in mind that it is difficult to determine whether profits explicitly stem from the new meal ordering system. In this regard, there are many examples where an information system can indeed account for profits. The case of the Jan/Feb 2023 supplementary exam focused on e-beats_Africa, a marketing company that specialises in promoting and monetizing TikTok artists. E-beats_Africa approaches Logic design to design an information system where thirdparty clients can purchase artist content or hire an artist to perform an act to be incorporated into their marketing/promotion campaigns. Logic design offer a design package that includes a 5-year subscription plan. Therefore, E-beats_Africa incurs annual expenses. E-beats_Africa, in turn, offers a monthly subscription plan to TikTok artist. Therefore, E-beats_Africa generates profit directly from the information system.

Reflect on the difference between the monetisation models that underlie E-beats_Africa and a solar panel system.

4. Adapt or fade? A case of e-hailing taxi services

Question 1.3 of Assessment 3 is an important question by virtue of its mark allocation. I would like to see thoughtful analysis and presentation of the critical assumptions and risks of Project X. To guide you, I discuss how the e-hailing cab services disrupted the taxi industry. I'd wager that South African metered taxi drivers, before 2013, did not expect that an app would upend the traditional metered taxi service. Two weeks ago, I flew from Cape Town to Joburg to attend a strategic planning seminar with the SoC (School of Computing). As soon as I disembarked the plane, I hailed an Uber taxi via the app. The process was hassle-free; no hailing, phoning, or googling; I know what car and driver to expect; I paid via Google wallet (had no cash on me). When I exited the terminal and entered the pick-up-drop-off area, I was stormed by metered taxi drivers offering to take me to my hotel; however, my Uber was already on its way. One cannot ignore that e-hailing is somewhat unregulated insofar as a significant profit flow out of the country that is in a struggling economy – Uber is an American transportation conglomerate. If you can't beat them, join them, right? However, metered taxi drivers have complained that Uber's required licensing fees are too expensive. Uber counter-argues that metered taxi drivers often do not meet road worthy criteria as vehicles are poorly maintained.

5. Sources consulted

Gilda, I. (2018). Arrow diagramming method example. Retrieved from ProjectCub!cle website: https://www.projectcubicle.com/arrow-diagramming-method-example/

Oguz, A. (2022). *Project management: Navigating the complexity with a systematic approach*. Cleveland, Ohio: MSL Academic Endeavors. Retrieved from https://pressbooks.ulib.csuohio.edu/project-management-navigating-the-complexity/chapter/7-4-creating-an-activity-network-diagram/

Schwalbe, K. (2019). *Information technology project management* (9th ed.). Boston, USA: Cengage Learning.

6. In closing

I hope this tutorial letter will help you to produce good solutions for Assessment 3. Please send me a mail if you have more questions.

Thank you and best wishes,

Emil Arthur van der Poll
Ph.D. student: Information Systems
School of Computing
vdpolae@unisa.ac.za

Enter Jiraiya’s honoured sage mode: The two great sage toads of Mount Myōboku!