Topological Sort and Critical Path Method on DAG for Optimal Marriage Routing in Harvest Moon: Back to Nature

Cendra Asih Chairunnisa - 13525017
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: cendra.asih.c@gmail.com , 13525017@std.stei.itb.ac.id

***Abstract*— Harvest Moon: Back to Nature is a classic farming simulation game in which one of the main objectives is to marry one of five candidates by raising the chosen candidate's affection and fulfilling additional event-based prerequisites. Because these prerequisites form a complex web of dependencies with varying in-game durations, the minimum time required to complete the marriage route is non-obvious, and players typically resort to lengthy trial-and-error. This paper models the marriage process for candidate Mary as a weighted Directed Acyclic Graph (DAG), where nodes represent in-game tasks, edges represent prerequisite relations, and weights represent task durations in in-game days. A topological ordering is applied to obtain a valid execution sequence, and the Critical Path Method (CPM) is used to compute the minimum completion time and identify the bottleneck tasks along the critical path. The result shows that the minimum time required to marry Mary is 212 in-game days, with the critical path consisting of the following sequence of activities: Start → Meet Mary → {Goddess Festival, Spring Thanksgiving, Horse Race Y1} → Gift Phase 1 → Purple Heart Event → Gift Phase 2 → Winter Thanksgiving → Blue Heart Event → Gift Phase 3 → Propose → Wedding Wait → Marry.**

Keywords—Harvest Moon, Marry Route, Directed Acyclic Graph, Topological Sort, Critical Path Method

I. Introduction

Harvest Moon: Back to Nature (HM:BTN) is a nostalgic farming simulation game where main gameplay is taking care grandfather's inherited unattended farm in Mineral Town. Main gameplays are growing crops, interacting with villagers, and marrying one of the five young ladies in town.

Fig. 1.1 *Harvest Moon: Back to Nature*, taken from [1]

Every candidate starts with black heart (acquaintances), then grow to purple, blue, green, yellow, orange, and final, red. The player also must give gift according her likes and act accordingly in several events to raise affection. In order to marry, the player have to expand their home twice size, reach heart level to red, and accepted after giving Blue Feather, sign of proposing. Marriage takes place one week after the player propose to chosen candidate.

Fig. 1.2 Marriage Candidates, Mary second from left, taken from [2]

Because of its many prerequisites that can be performed in parallel while others are mutually locking, the order in which activities are executed affects the total in-game time required to reach marriage. A player seeking to optimize their marriage route faces a classical project scheduling problem, raising questions, which activities determine the minimum duration, and which possess slack and can therefore be rescheduled without delaying the overall completion.

This paper models the marriage prerequisites of HM:BTN as a Directed Acyclic Graph (DAG) and applies two concepts from discrete mathematics, Topological Sort and Critical Path Method (CPM), in order to determine a valid execution order of activities and compute the minimum marriage duration.

II. Theoretical Foundation

A. Harvest Moon: Back to Nature Marriage Candidate

In Harvest Moon: Back to Nature, every marriage candidate begins the game with a zero affection point (AP). Affection accumulates over the course of the game and is bounded by an upper limit of 65.535 AP. The continuous AP value is partitioned into seven discrete hearts as listed,

Heart Color	Affection
Red	60,000 - 65,535
Orange	50,000 - 59,999
Yellow	40,000 - 49,999
Green	30,000 - 39,999
Blue	20,000 - 29,999
Purple	5,000 - 19,999
Black	0 - 4,999

Tab. 2.1 Heart Level Point Range, taken from [9]

Heart levels serve as prerequisites for triggering the candidate's heart events and unlocking the marriage option. Affection is raised primarily by giving gifts, with the AP earned per gift depending on several factors, listed below

	Girl	if wrapped	on Girl's birthday	if wrapped	Villager	if wrapped	on Villager's birthday	if wrapped
Loved	500/300	500/500	500/300	2500/1500	9/3	11/2	45/15	47/40
Liked	300/100	300/100	300/100	1500/500	3/1	5/0	15/5	17/0
Neutral	100/0	100/0	100/0	500/0	1/0	3/-1	5/0	7/-5
Disliked	-500/-500	-500/-500	-500/-500	-2500/-2500	-3/-3	-1/-4	-15/-15	-13/-20
Hated	-800/-800	-800/-800	-800/-800	-4000/-4000	-9/-9	-7/-10	-45/-45	-43/-50
Perfume (can't wrap)	+1000	--	+1000	--	--	--	--	--

Tab. 2.2 Gift Gifting AP, taken from [3]

Additional AP are obtained through several recurring in-game events, such as daily conversation, attending festivals together, and triggering the candidate's heart events. Since the present paper focuses on Mary as the marriage route under analysis, the AP contributions associated with each of these events are detailed in Section III.

B. *Mary*

Fig. 2.1 Mary Charachter, taken from [4]

Mary is one of the five marriage candidates in *Harvest Moon: Back to Nature*. She lives at the Mineral Town Library. Her birthday is Winter 20. Mary's *Loved* gifts include items closely tied to her father's botanical work, in particular the Moondrop Flower, the Pink Cat Flower, and the Toy Flower, as well as several wild herbs and mountain forageables, while she dislikes most animal products and mineral items.

C. *Directed Acyclic Graph (DAG)*

A graph is an ordered pair G = (V, E), where V is a finite set of vertices and E is a set of edges connecting pairs of vertices. A graph is called directed (digraph) if every edge has an orientation, where each edge is an ordered pair $(u, v) \in V \times V$ representing a connection from u to v. A directed path of length k from vertex v_0 to vertex v_k is a sequence of vertices v_0, v_1, …, v_k such that $(v_{i-1}, v_i) \in E$ for all $1 \leq i \leq k$. A directed cycle is a directed path in which $v_0 = v_k$ and $k \geq 1$.

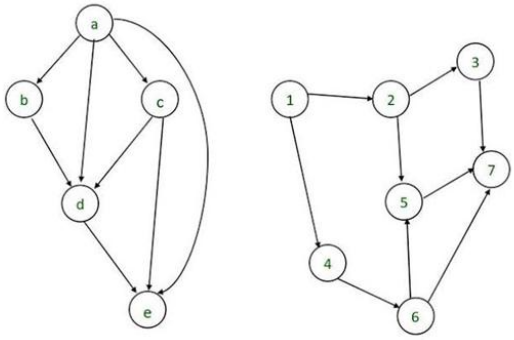


Fig. 2.2 Directed Acyclic Graph, taken from [10]

A Directed Acyclic Graph (DAG) is a directed graph that contains no directed cycles. DAGs structure represent precedence relation where an edge (u, v) indicates that u must occur, or be processed, before v. Because of acyclicity, it guarantees that the system can be executed in finite time.

In the context of Harvest Moon: Back to Nature, the marriage process for a chosen candidate is modeled as a weighted DAG G = (V, E, w), where V is the set of in-game activities required to reach the marriage event, $E \subseteq V \times V$ encodes the prerequisite relation between activities (an edge $(u, v) \in E$ means activity u must be completed before activity v can begin), and $w : V \rightarrow \mathbb{R}^+$ assigns each activity its estimated duration in in-game days.

D. *Topological Sort*

A topological ordering of a directed graph G = (V, E) is a linear ordering of its vertices $v_1, v_2, \ldots, v_n$ such that for every edge $(v_i, v_j) \in E$, the relation $i < j$ holds. Intuitively, every vertex appears in the ordering before all of its successors.

Based on fundamental theorem, a directed graph G admits a topological ordering if and only if G is acyclic.

If G contains a cycle $v_0, v_1, \ldots, v_k = v_0$, then no ordering can place v_0 before all its successors while also placing each successor before v_0. The "if" direction can be proved constructively via Kahn's algorithm

Kahn's algorithm runs in $O(|V| + |E|)$ time. A given DAG generally admits multiple valid topological orderings, the number of valid orderings reflects the degree of scheduling freedom in the underlying system.

In the marriage routing problem, a topological ordering corresponds to a valid execution sequence of activities. Any ordering produced by topological sort respects every prerequisite and therefore represents a feasible play-through. However, topological sort alone does not select among feasible orderings, it neither minimizes total duration nor identifies bottlenecks. For that, comes the Critical Path Method.

E. *Critical Path Method*

The Critical Path Method (CPM) on a weighted DAG in which each vertex represents an activity with a known

duration, and edges represent precedence constraints, the method computes the minimum total duration required to complete the entire project and identifies the sequence of activities that determines this minimum.

For each activity $v \in V$, CPM defines four quantities:

1. The earliest start time ES(v): the earliest in-game day on which activity v can begin.
2. The earliest finish time $EF(v) = ES(v) + w(v)$.
3. The latest finish time LF(v): the latest day on which v can finish without delaying the project.
4. The latest start time $LS(v) = LF(v) - w(v)$.

These quantities are computed in two passes over the DAG, which both require the vertices to be processed in topological order:

Forward pass (computing ES and EF). For every source vertex v, set $ES(v) = 0$. For every other vertex v,

$ES(v) = \max\{EF(u):(u,v) \in E\}$,

$EF(v) = ES(v) + w(v)$.

The project's minimum total duration is then

$$T^* = \max\{ EF(v) : v \text{ is a sink } \}$$

Backward pass (computing LF and LS). For every sink vertex v, set $LF(v) = T^*$. For every other vertex v,

$LF(v) = \min\{ LS(u) : (v, u) \in E \}$,

$LS(v) = LF(v) - w(v)$.

The slack (or *float*) of an activity is defined as

$$slack(v) = LS(v) - ES(v) = LF(v) - EF(v)$$

An activity v with $slack(v) = 0$ is called critical, any delay in its execution directly delays the entire project. The critical path is the sequence of critical activities connecting a source to a sink, this path has total weight T^* and forms the project's bottleneck. Activities with positive slack possess scheduling flexibility and may be deferred by up to slack(v) days without delaying the marriage event.

Both passes run in $O(|V| + |E|)$ time, so the entire CPM analysis, including the preceding topological sort, is linear in the size of the DAG.

In the marriage routing problem, T^* represents the minimum in-game time required to marry the chosen candidate, the critical path identifies the bottleneck chain of activities, and the slack values indicate which activities (e.g. optional gift-giving days or non-prerequisite heart events) may be rescheduled without lengthening the route.

III. Data Preparation

A. *Mary Affecionate Point*

This table provide each act the player can do along with earned point from interacting with Mary

Requirement	Action	Point Earned (point)
Introduction	Ask Mary what she is writing upon first meeting her	2000
-	Give Mary a gift	Varies (see Table 2.2)
Purple Heart	Choose the "came to read" dialogue option when visiting the library	2000
Blue Heart, after Winter Thanksgiving	Mary lends the book *The Woodcutter and the King*; the player selects the dialogue option preferring the Woodcutter character	2000
New Year's Day	Talk to Mary	500
Goddess Festival & Purple Heart	Talk to Mary only	500
	Accompany Mary through the festival	1000
	Accompany Mary through the festival, then visit her favorite place	3000
Thanksgiving Festival	Give Mary a sweet she likes (AP awarded only for the first gift)	1000
Spring Festival	Talk to Mary	500
Horse Race	Talking with Mary (up to 3 times)	500 per talk
	Win the race	500
Cooking Festival	Talk to Mary	500
	Win the contest	500
Swimming Festival	Talk to Mary	500
	Win the festival	500
Tomato Festival	Talk to Mary	500
	Win as a team (Mary must not be on the player's team)	500
Cow Festival	Talk to Mary	500
	Win the contest	500
Fireworks Display	Talk to Mary	500
Music Festival	Talk to Mary	500
Harvest Festival	Talk to Mary	500
Sheep Festival	Talk to Mary	500

	Win the contest	500
New Years Eve	Talk to Mary	500

Tab. 3 Mary Affection Point Table

This paper models all in-game activities required to marry Mary as a set of weighted nodes, defines the prerequisite edges between them, and weight (duration in in-game days) of each node. All durations are estimated under the following assumptions:

1. The player gives Mary a loved gift twice per week (2 × 800 AP = 1,600 AP per 7 days),
2. The player attends every available festival and selects the dialogue option that maximizes AP, and
3. Each House Upgrade is commissioned immediately upon reaching the required funds.

IV. DAG Model

A. DAG Node

The marriage process is modeled as a weighted DAG G = (V, E, w). Table 4.1 lists every node, its weight w(v) in in-game days with a brief description.

ID	Node	Weight (days)	Notes
A	Start	0	Day 1
B	Meet Mary	1	Day 1, +2000 AP
C	Gift Giving Phase 1	19	Day 1–20, reach Purple Heart
D	Goddess Festival	1	Day 14
E	Spring Thanksgiving	1	Day 18
F	Horse Race Y1	1	Day 22
G	Purple Heart Event	1	~Day 22, library dialogue
H	Gift Giving Phase 2	33	Day 22–55, reach Blue Heart
I	Blue Heart prerequisite: Winter Thanksgiving	1	Day 114
J	Blue Heart Event	1	Day 115, +2000 AP
K	Gift Giving Phase 3	89	Day 55–205, reach Red Heart
L	Festivals Y1 (Tomato, Swimming, Cooking, Cow, Harvest, Music, Sheep)	varies	parallel with gifting
M	Festivals Y2 (Horse Race, Fireworks, Tomato)	varies	parallel with gifting
N	Earn Money Phase 1	30	House Upgrade 1
O	House Upgrade 1	3	Construction 3 days
P	Earn Money Phase 2	60	House Upgrade 2
Q	House Upgrade 2	3	Construction 3 days
R	Won Arrives	1	Day 8 (fixed)
S	Raise Horse (1 year)	120	Day 8 → Day 128
T	Buy Blue Feather	1	Year 2 (Day 121+)
U	Propose (give Blue Feather)	1	Prerequisite after R+H+Q+T done
V	Wedding Wait	7	7 after propose
W	Marry	0	sink node

Tab. 4.1 Node Table

B. DAG Edge List

Table 4.2 lists all directed edges (u, v) ∈ E, where (u, v) means activity u must be completed before activity v can begin.

Pathway	Description
A → B	Start → Meet Mary
A → N	Start → Earn Money 1 (parallel)
A → R	Start → Won Arrives (fixed day 8)
B → C	Meet Mary → Gift Phase 1
B → D	Meet Mary → Goddess Festival (day 14)
B → E	Meet Mary → Spring Thanksgiving (day 18)
B → F	Meet Mary → Horse Race Y1 (day 22)
D → C	Goddess Festival contributes to AP (in phase 1)
E → C	Spring Thanksgiving contributes AP
F → G	Horse Race → Purple Heart Event possible
C → G	Purple Heart → Purple Heart Event
G → H	Purple Heart Event → Gift Phase 2
H → I	Blue Heart → Winter Thanksgiving possible
I → J	Winter Thanksgiving → Blue Heart Event
J → K	Blue Heart Event → Gift Phase 3
H → L	Blue Heart reached → Festivals Y1 (parallel)
K → M	Y2 festivals parallel with gifting
L → K	Festival flow back to Phase 3
M → K	Festivals contribute AP to Phase 3
K → U	Red Heart → Can propose

Tab. 4.2 Edge Table

C. DAG Model

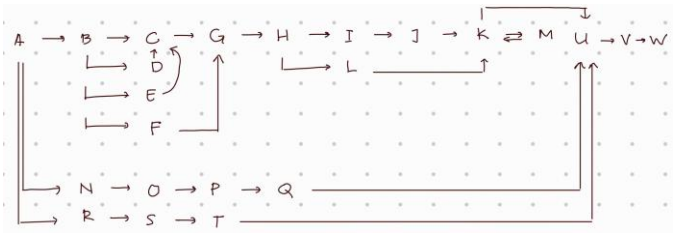


Fig. 4 DAG Model

V. Analysis

This section applies topological sort and the Critical Path Method (CPM) to the DAG defined in Section III. The result

is the minimum number of in-game days required to marry Mary and the identification of the critical path.

A. Topological Ordering

A topological ordering of G is a linear sequence of all nodes such that for every edge (u, v), node u appears before node v. Applying Kahn's algorithm to the DAG defined in Section III gives one the following valid ordering:

A → R → N → B → D → E → F → C → O → G → P → H → Q → I → L → J → S → K → T → M → U → V → W

B. CPM Forward Pass

The forward pass computes the Earliest Start time ES(v) and Earliest Finish time EF(v) for each node, processing nodes in topological order. For every source node, ES = 0. For all other nodes, ES(v) = max{EF(u):(u, v) ∈ E}, and EF(v) = ES(v) + w(v).

Node	Activity	w(v)	ES(v)	EF(v)
A	Start	0	0	0
B	Meet Mary	1	0	1
D	Goddess Festival	1	1	2
E	Spring Thanksgiving	1	1	2
F	Horse Race Y1	1	1	2
C	Gift Phase 1	19	1	20
G	Purple Heart Event	1	20	21
H	Gift Phase 2	33	21	54
I	Winter Thanksgiving	60	54	114
J	Blue Heart Event	1	114	115
L	Festivals Y1	—	54	—
K	Gift Phase 3	89	115	204
M	Festivals Y2	—	116	—
N	Earn Money 1	30	0	30
O	House Upgrade 1	3	30	33
P	Earn Money 2	60	33	93
Q	House Upgrade 2	3	93	96
R	Won Arrives	1	8	9
S	Raise Horse	120	9	129
T	Buy Blue Feather	1	129	130
U	Propose	1	204	205
V	Wedding Wait	7	205	212
W	Marry	0	212	212

Tab. 4.3 Forward pass results (ES and EF)

The minimum total duration is T* = EF(W) = 212 in-game days.

C. CPM Backward Pass

The backward pass computes the Latest Start time LS(v) and Latest Finish time LF(v), processing nodes in reverse topological order. For every sink node, LF = T*; for all other nodes, LF(v) = min{LS(u):(v, u) ∈ E} and LS(v) = LF(v) − w(v). The slack of each node is defined as slack(v) = LS(v) − ES(v).

ID	Activity	ES	EF	LS	LF	Slack	Critical?
A	Start	0	0	0	0	0	Yes
B	Meet Mary	0	1	0	1	0	Yes
C	Gift Phase 1	1	20	1	20	0	Yes
D	Goddess Festival	1	2	1	2	0	Yes
E	Spring Thanksgiving	1	2	1	2	0	Yes
F	Horse Race Y1	1	2	1	2	0	Yes
G	Purple Heart Event	20	21	20	21	0	Yes
H	Gift Phase 2	21	54	21	54	0	Yes
I	Winter Thanksgiving	54	114	54	114	0	Yes
J	Blue Heart Event	114	115	114	115	0	Yes
K	Gift Phase 3	115	204	115	204	0	Yes
L	Festivals Y1	54	—	54	—	> 0	No
M	Festivals Y2	204	—	116	—	>0	No
N	Earn Money 1	0	30	108	138	108	No
O	House Upgrade 1	30	33	138	141	108	No
P	Earn Money 2	33	93	141	201	108	No
Q	House Upgrade 2	93	96	201	204	108	No
R	Won Arrives	8	9	82	83	74	No
S	Raise Horse	9	129	83	203	74	No
T	Buy Blue Feather	129	130	203	204	74	No
U	Propose	204	205	204	205	0	Yes
V	Wedding Wait	205	212	205	212	0	Yes
W	Marry	212	212	212	212	0	Yes

Tab. 4.3 Backward pass results and slack values

D. Critical Path

An activity v is critical if slack(v) = 0. The critical path is:

A → B → D/E/F → C → G → H → I → J → K → U → V → W

This path has total weight T* = 212 days and represents the minimum time required to marry Mary.

VI. Conclusion

These results show that the minimum time required to marry Mary, under the stated assumptions, is 212 in-game days. The critical path runs entirely through the affection accumulation activities, with the single most impactful scheduling constraint being the calendar-locked Blue Heart Event (node J), whose prerequisite (Winter Thanksgiving, node I) cannot occur before Day 114 regardless of how quickly the player satisfies the Blue Heart AP threshold.

Video Link at Youtube

https://youtu.be/IUnPHGRCj1M

Acknowledgment

The author would like to thank, first of all, to God for all the guidance throughout the process of learning thus writing the contents of this paper. The author would also like to thank

the lecturer of ITB Discrete Mathematics IF2120, Mr. Rinaldi Munir for sharing their knowledges and guide the students throughout the learning process in the class. And especially, for the people who keep giving motivation to the author in the midst of their own busy life, the author would like to give special gratitude to her family and friends who have given the support throughout the entire semester.

REFERENCES

[1] Natsume Inc., "Harvest Moon: Back to Nature," Facebook, [Online]. Available: https://www.facebook.com/NatsumeInc. [Accessed: Jun. 19, 2026].

[2] "Harvest Moon: Back To Nature," *Harvest Moon Wiki*, Fandom. [Online]. Available: https://harvestmoon.fandom.com/wiki/Harvest_Moon:_Back_To_Nature. [Accessed: Jun. 19, 2026].

[3] M. Vinson, "Harvest Moon: Back To Nature — Marriage Guide," *GameFAQs*. [Online]. Available: https://gamefaqs.gamespot.com/ps/446412-harvest-moon-back-to-nature/faqs/13757. [Accessed: Jun. 19, 2026].

[4] "Mary (BTN)," *Harvest Moon Wiki*, Fandom. [Online]. Available: https://harvestmoon.fandom.com/wiki/Mary_(BTN). [Accessed: Jun. 19, 2026].

[5] "Mary (Harvest Moon: Back to Nature)," *Ranch Story Wiki*, Miraheze. [Online]. Available: https://ranchstory.miraheze.org/wiki/Mary_(Harvest_Moon:_Back_to_Nature). [Accessed: Jun. 19, 2026].

[6] "Harvest Moon: Back to Nature — Girls," *Harvest Moon Back to Nature Guide*. [Online]. Available: https://www.harvestmoonbacktonatureguide.com/girls.html. [Accessed: Jun. 19, 2026].

[7] "HMBTN" [Harvest Moon: Back to Nature guide], *Scribd*. [Online]. Available: https://www.scribd.com/document/384566639/hmbtn. [Accessed: Jun. 19, 2026].

[8] Victor Interactive Software, *Harvest Moon: Back to Nature*. Tokyo, Japan: Natsume, 1999.

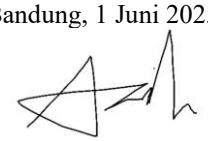
[9] "Affection (BTN)," *Harvest Moon Wiki*, Fandom. [Online]. Available: https://harvestmoon.fandom.com/wiki/Affection_(BTN). [Accessed: Jun. 19, 2026].

[10] GeeksforGeeks, "Directed Acyclic Graph in Compiler Design with Examples," GeeksforGeeks, Jul. 23, 2025. [Online]. Available: https://www.geeksforgeeks.org/compiler-design/directed-acyclic-graph-in-compiler-design-with-examples/. [Accessed: Jun. 19, 2026].

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 1 Juni 2025



Cendra Asih Chairunnisa / 13525017