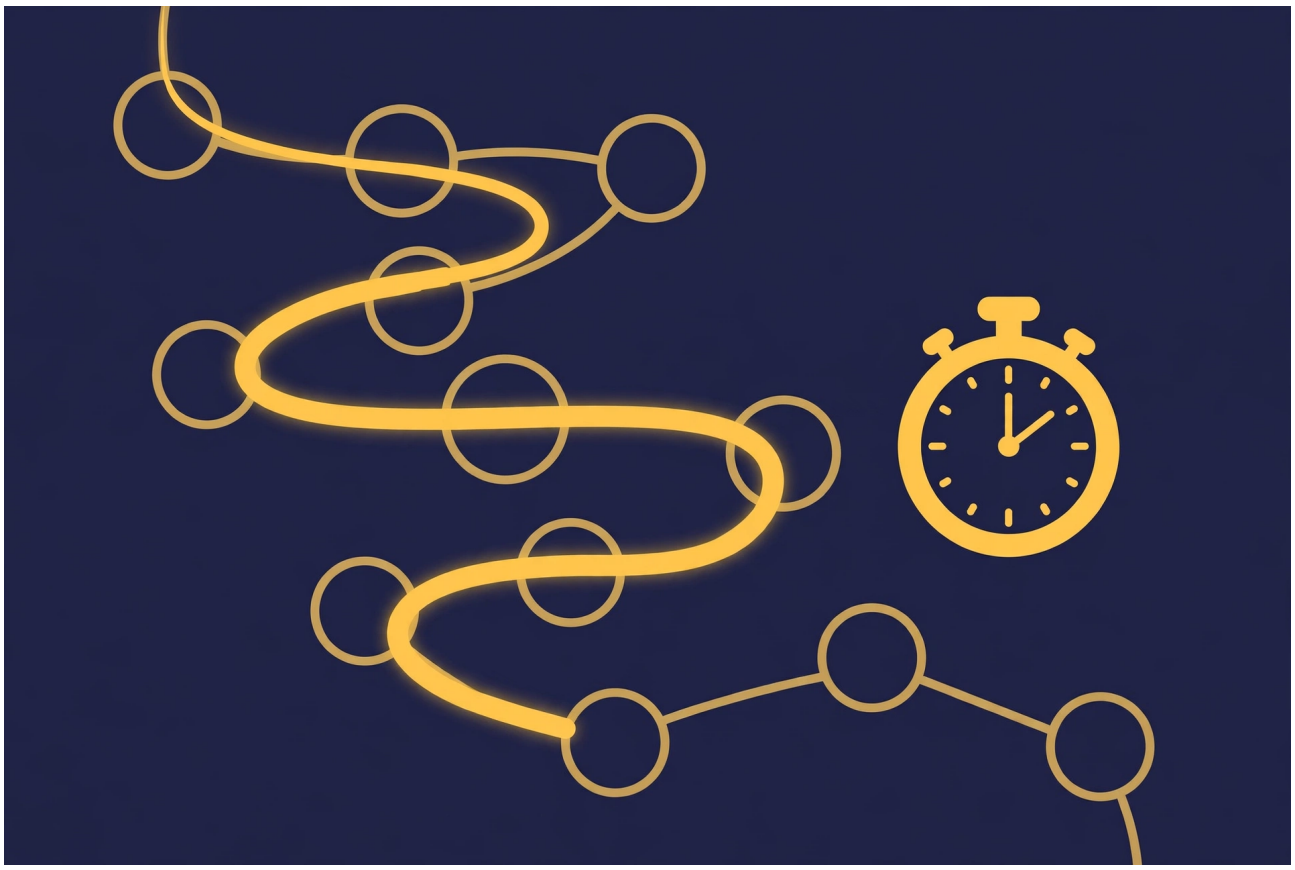
By Bright Quill Team



A critical path method example is worth more than a page of definitions. The method only makes sense when you watch it work on real tasks. It's simpler than it sounds. The critical path method is a scheduling technique that finds the longest chain of dependent tasks in a project, and that chain sets the shortest possible finish date. Every project has one. Tasks on the critical path can't slip without pushing the deadline. Tasks off it usually have float, which is spare time you can use flexibly. In this article you'll work through a simple example with six tasks, see the forward and backward pass step by step, and learn float in words that actually stick.

What the Critical Path Method Does

Every project is a web of tasks with dependencies. You can't paint walls before the plaster dries. The critical path method maps those dependencies and calculates dates for every task. The longest path through the network is the critical path. Its length is the shortest time the project can finish in, because every task on that path must happen in sequence. Tasks on the critical path have zero float. Delay one by a day and the whole project slips by a day. Tasks off the critical path have float, which means they can start late or take longer without hurting the finish date. Smart managers watch the critical path like a hawk. They'll use float to juggle resources elsewhere. Our business plans team runs this calculation on every client schedule. A plan with no critical path is just a wish list.

The Example: Six Tasks to Build a Garden Shed

Let's build a garden shed. It's a fictional weekend project with six tasks, durations and dependencies. Task A, clear and level the ground, takes 2 days, no dependencies. Task B, pour the concrete base, takes 3 days,

depends on A. Task C, order and deliver the shed kit, takes 5 days, no dependencies. It can run in parallel with the groundwork. Task D, assemble the shed walls and roof, takes 2 days, depends on B and C. Both the base and the kit must be ready. Task E, paint the shed, takes 2 days, depends on D. Task F, build the shelving inside, takes 1 day, depends on D, and it can run in parallel with painting. Draw these as boxes with arrows. That network diagram is the starting point for the whole method.

The Forward Pass: Earliest Dates

The forward pass walks the network from start to finish asking one question: what's the earliest each task can start and finish. Start at day 0. Task A runs days 0 to 2. Task C runs days 0 to 5. Task B needs A finished, so it runs days 2 to 5. Task D needs both B and C. B finishes day 5, C finishes day 5, so D starts day 5 and runs to day 7. Task E needs D, so it runs days 7 to 9. Task F needs D too, so it runs days 7 to 8. The project finishes at day 9, set by task E. Write two numbers on each box: earliest start and earliest finish. Don't skip this. That's the forward pass done. It tells you the fastest the project can go if everything starts as early as possible.

The Backward Pass: Latest Dates and Float

The backward pass walks from finish to start asking the opposite question: what's the latest each task can start without delaying the project. The project must finish day 9, so task E, which ends the project, has latest finish 9 and latest start 7. Task F must also finish by 9, so its latest start is 8. Task D feeds both E and F. E needs D done by 7, F needs D done by 8, so D must finish by the earlier date, day 7, giving latest start 5. Task C feeds D, so C must finish by 5, latest start 0. Task B feeds D, so B must finish by 5, latest start 2. Task A feeds B, so A must finish by 2, latest start 0. Now compare. Tasks where earliest and latest dates match have zero float: A, B, C, D, E. Task F has earliest start 7 but latest start 8, so it has 1 day of float. The critical path is A, B, C, D and E. The project takes 9 days. There's no way around it.

Reading the Answer Like a Project Manager

Three things fall out of the calculation. First, the finish date: 9 days. No amount of clever scheduling beats it. You'd have to change the work itself. Second, the watch list: tasks A, B, C, D and E. If the shed kit delivery slips, the project slips, because C sits on the critical path. Third, the flexibility: task F has a day of float, so the shelving can wait a day if the painter needs help. That's how managers use the method in real life. They protect critical tasks, sometimes adding resources to shorten them, which is called crashing. They move float tasks around to smooth the workload. Our planning experts at Bright Quill highlight the critical path in every schedule we build, so you'll see exactly where the risk sits.

Five Critical Path Mistakes Beginners Make

Here's where it usually goes wrong. First, forgetting a dependency. Miss one arrow and the whole network lies. List dependencies before drawing anything. Second, confusing the longest path with the most tasks. The critical path is the longest in duration, not in task count. Third, treating float as free time to waste. Float is a buffer for problems, not an excuse to start late. Fourth, calculating once and never updating. Don't. Real projects change, and the critical path can shift to a different chain mid project. Recalculate at every progress review. Fifth, presenting the answer without the diagram. Markers want to see the network with earliest and latest dates on each node. The numbers prove you've actually done the method instead of guessing.

A critical path method example like the shed build shows the whole technique: map the dependencies, run the forward pass for earliest dates, run the backward pass for latest dates, and read off the path with zero float. Try it on your own project. You'll find scheduling stops feeling like guesswork. If your assignment needs

a worked critical path with clean diagrams explained, reach us through the contact page .

Key visual, in brief:

Example: A clear site (3 days), then B build frame (2 days), then D roof (4 days) gives a 9 day critical path. C paint (5 days) runs in parallel with 4 days of float.