

## Sequencing: Troubleshoot a Process Automation System

### Sequencing strategy

To determine the sequence of modules and units, I first gathered the following type of data about each unit of behavior:

Consideration	Details
Difficulty to learn	<p>Difficulty considers these characteristics of the behavior in the unit:</p> <ul style="list-style-type: none"> <li>▪ Number of steps</li> <li>▪ Fineness of judgment needed to tell things apart</li> <li>▪ Number of judgments to tell things apart</li> <li>▪ Range and exceptions to any rules</li> <li>▪ Speed or tolerances needed</li> </ul> <p>The more of those characteristics that a behavior has, the more difficult it is to learn. To compare behaviors for the purpose of sequencing units, I classified behavior as low, medium, or high in difficulty to learn.</p>
Competition	<p>Learning one behavior interferes with learning another because the situations are similar but require different actions.</p>
Recurring sub-skill	<p>Sometimes, several behaviors share a common sub-set of actions or a common concept. For example, you use the concept of “righty, tifty; lefty, loosey” whether you are changing a light bulb, fixing a flat tire, or connecting the cable to your cable box. So we can consider it a sub-skill in the context of those behaviors.</p> <p>It’s usually more efficient to teach sub-skills before the behaviors that use them. Then, when you teach the behavior, you just have to say the name of the sub-skill, such as “Tighten the cable.”</p>
Job order	<p>Job order considers the order in which things are normally done on the job. Typically, certain behavioral processes are first to produce the results or outputs needed for subsequent processes.</p>

By gathering the above data, I could apply these rules to determine the sequence that would best support students' learning:

	<b>Rule</b>	<b>Reason</b>
1	Teach difficult behavior early	This gives students more opportunity to concentrate on and practice the difficult behavior. There is also less interference from other behaviors.
2	Teach sub-skills before the behavior that requires them.	Subsequent behavior will be easier to learn because students already have a beginning language and set of skills to draw upon.
3	Teach competing behavior in close proximity to each other.	This gives students a chance to compare and contrast the behaviors.
4	If none of the above applies, teach in job order.	Correct perform often depends on doing things in the right order.

### Interviews with accomplished technicians

To gather data to make the sequence decisions, I talked with technicians who are accomplished in troubleshooting process automation systems. They provided information on how to perform each unit of behavior. They also provided the following information on the general job order of the units and which ones they do most often:

- When they troubleshoot a problem, they first do a series of isolation actions to narrow down where the problem is likely to be. After that, they do diagnostic checks to pinpoint the problem.
- About 90% of the time, they are troubleshooting problems with field devices such as valves, motors, and pumps.
- Problems with workstations and servers are less common. They only comprise about 1 out of every 10 problems.

## Sequencing results

The remaining sections summarize the sequence of modules and units for the course, along with the data supporting the sequence. The data is based on the interviews with the accomplished technicians and analysis of the steps required to perform each unit.

### Module 1: Find and trace signals through function blocks

The first module comprises a set of sub-skills students will use in many of the other units. By teaching these sub-skills first, the students will have a foundation to move more efficiently through the rest of the course:

Unit		Difficulty			Interactivity			Notes
		L	M	H	None	Sub-skill	Competes	
1.1	Prepare equipment for restart	x				x		Sub-skill done as the last step after resolving most problems.
1.2	Find data online		x			x		Sub-skill used when you isolate and diagnose various problems
1.3	Trace through a function block diagram		x			x		Sub-skill used when you isolate and diagnose various problems
1.4	Find I/O chassis, slot, and terminals for device	x				x		Sub-skill used when you isolate and diagnose various problems

## Module 2: Troubleshoot discrete device alarms

Module 2 covers a set of skills that rank high in difficulty. They are also the types of problems that occur most often. By teaching them first, students will have a chance to revisit them throughout the course and strengthen those skills:

Unit		Difficulty			Interactivity			Notes
		L	M	H	None	Sub-skill	Competes	
2.1	Isolate alarm condition		x		x			
2.2	Isolate problem with field device			x	x			
2.3	Diagnose problem with discrete output device			x	x			
2.4	Diagnose problem with discrete input device			x	x			

## Module 3: Troubleshoot I/O faults

Module 3 includes several units that ranked high in difficulty. It also includes a sub-skill, but students will only use that skill within this module, so I included it here:

Unit		Difficulty			Interactivity			Notes
		L	M	H	None	Sub-skill	Competes	
3.1	Isolate I/O fault		x		x			
3.2	Find I/O channel data		x			x		Sub-skill used when you isolate and diagnose analog problems
3.3	Diagnose problem with analog input device			x	x			
3.4	Diagnose problem with analog output device			x	x			
3.5	Diagnose module problem	x			x			
3.6	Diagnose EtherNet/IP problem		x		x			

## Module 4: Troubleshoot non-alarm problems

Module 4 covers device-type problems that are similar to the previous modules but are less difficult:

Unit		Difficulty			Interactivity			Notes
		L	M	H	None	Sub-skill	Competes	
4.1	Isolate non-alarm problem		x		x			
4.2	Diagnose permissives and interlocks	x			x			
4.3	Diagnose device mode problem		x		x			
4.4	Bypass devices		x		x			

## Module 5: Troubleshoot switchovers and operator workstation problems

Module 5 teaches the units of behavior that are used only about 10% of the time. Although some of the units rank high in difficulty, I left this module for last because of its low frequency of occurrence on the job. I also included diagnosing switchovers because of its competition with server problems. Both units use similar terms but the terms have different meanings in each unit:

Unit		Difficulty			Interactivity			Notes
		L	M	H	None	Sub-skill	Competes	
5.1	Diagnose switchover		x				x	Terms used different between servers and controllers.
5.2	Isolate operator workstation problem			x	x			
5.3	Check server status		x		x			
5.4	Ping servers and workstations	x				x		Sub-skill used when you isolate a server problem
5.5	Isolate server problem			x			x	Terms used different between servers and controllers.