1. Do you have a current, credible activity network supported by a Work Breakdown Structure?
   ■ Have you identified your critical path items?
   ■ What explicit provisions have you made for work that isn’t on your WBS?
   ■ Does the activity network clearly organize, define, and graphically display the work to be accomplished?
   ■ Does the top-level activity network graphically define the project from start to finish, including all the dependencies?
   ■ Does the lowest-level WBS show work packages the measurable tasks and short duration?
   ■ Does each task on the network have a well-defined deliverable?
   ■ Is each work package under budget control, expressed in labor hours, dollars, or other numerical units?

2. Do you have a current, credible schedule and budget?
   ■ Is the schedule based on a project/activity network supported with a WBS?
   ■ Is the schedule based on realistic, historical, quantitative performance estimates?
   ■ Does the schedule provide time for holidays, vacation, sick days, etc?
   ■ Does the schedule allow all the necessary quality assurance activities?
   ■ Does the schedule account for resource overlaps?
   ■ Is the schedule for the three to six months as detailed as possible?
   ■ Is the schedule consistently updated at all levels in the Gantt, PERT, and Critical Paths views?
   ■ Can you perform to the schedule and budget?

3. Do you know what software you are responsible for delivering?
   ■ Are the system’s operational requirement clearly specified?
   ■ Are definitions of what the software must do to support the system operational requirements clearly specified?
   ■ Are system interfaces clearly specified and if appropriate prototyped?
   ■ Is the selection of the software architecture and design method traceable to system operational characteristics?
   ■ Are descriptions of the system environment and relationships of software application to the system architecture clearly specified?
   ■ Are specific development requirements explicitly defined?
   ■ Are specific acceptance and delivery requirements explicitly defined?
   ■ Are user requirements agreed to by joint teams of developers and users?
   ■ Are system requirements traceable through the software design?

4. Can you list the current top ten project risks?
   ■ Has a Risk Management role been assigned to the project?
   ■ Are risks determined through established processes for risk identification, assessment, and mitigation?
   ■ Is there a database that includes all non-negligible risks in terms of probability, earliest expected visible symptom, estimated and actual schedule, and cost effects?
   ■ Are all project personnel encouraged to become risk identifiers?
   ■ Is the database of top-ten risks updated regularly?
   ■ Are user requirements reasonably stable?
   ■ Do you know how the risks are changing over time?

5. Do you know your schedule compression percentage?
   ■ Has the schedule bee constructed bottom up from quantitative estimates, not by predetermined dates?
   ■ Has the schedule been modified when major modifications in the software have taken place?
   ■ Have developers and test personnel received training in the principle domain area, support software and tools?
   ■ Have detailed unit-level and interface design specifications been created for maximum parallel developer effort?
   ■ Does the project avoid extreme dependencies on specific individuals?
   ■ Are people working abnormal hours?
   ■ Do you know the historical schedule compression percentage on similar projects, and the results of those projects?
   ■ Is any part of the schedule compression based on use of new technologies?
   ■ Has the percentage of software functionality been decreased in proportion to the percentage of schedule compression?

6. What is the estimated size of your software deliverables?
   ■ Has the project scope been clearly established?
   ■ Were measurements from the previous projects used as a basis for size estimates?
   ■ Were Source Lines of Code (SLOC) used as a basis for the estimates?
   ■ Were Function Points (FP) used as a basis for estimates?
   ■ Are the developers who do the estimating experienced in the domain area?
   ■ Are estimates regularly updated to reflect software development realities?

7. Do you know the percentage of external interfaces that are not under your control?
   ■ Has each external interface been identified?
   ■ Have critical dependencies of each external interface been documented?
   ■ Has each external interface been ranked based on potential project impact?
   ■ Have procedures been established to monitor external interfaces until the risk is eliminated to substantially reduced?
   ■ Have agreements with the external interface controlling organizations been reached and documented?

8. Does your staff have sufficient expertise in the project domain?
   ■ Do you know what the users needs, wants, and expects?
   ■ Does the staffing plan include a list of the key expertise areas and estimated number of personnel needed?
   ■ Does most of the project staff have experience with the specific type of system being developed?
   ■ Does most of the project staff have extensive experience in the software language being used?
   ■ Are the developers able to proceed without undue request for additional time and cost to help resolve technical problems?
   ■ Are the developers knowledgeable in domain engineering – the process of choosing the best model for the project and using it throughout design, code, and test?
   ■ Is there a domain expert assigned to each domain?

9. Have you identified adequate staff to allocate to the scheduled tasks at the right time?
   ■ Do you have sufficient staff to support the tasks identified in the activity network?
   ■ Is the staffing plan based on historical data of level of effort to staff months on similar projects?
   ■ Do you have adequate staffing for the current tasks and all tasks scheduled to occur in the next two months?
   ■ Have alternative staff buildup approaches been planned?
   ■ Does the staff buildup rate match the rate at which the project leaders identify unsolved problems?
   ■ Is there sufficient range and coverage of skills on the project?
   ■ Is there adequate tile allocated for staff vacations, sick leave, training, and education?
   ■ Are staffing plans regularly updated to reflect reality?