Exercise 3 – Software Engineering Economics

1. Each project is grouped into one of four groups, according to their size (expressed in function points). The performance of the projects in the repository is measured in productivity, time-to-market and process quality, and is related to the size of a project.

   For each project the measure of deviation from the average trend line (of each size group) is calculated and expressed in a percentage; negative when below the average trend line and positive when above the trend line. Based on this percentage all projects from the repository are plotted in a matrix, resulting in the four cost/duration quadrants.

   The projects that score as good practice on average shows the best productivity, time-to-market and process quality of all four quadrants. For bad practices these performance indicators are the lowest of all four quadrants.

   When aspects of the projects are strongly related (50% or more) to a high percentage of good practice the aspects are success factors, when they are strongly related (50% or more) to a high percentage of bad practice, the aspects are failure factors.

2. It is difficult to change the Programming Language to Visual Basic (it is easier to find improvements by using the success factors 1-4 in the paper). Software companies cannot just change their business organization structure and trying to switch to another programming language asks for a long term approach.

3. - Project within one office of a company (office where all the team members work)
   This would belong to good practice because the team would be able to have meetings every day and communicate with each other on a daily basis. This will help speed up the process to achieve the end-product.

   - International, world-wide team
   This would belong to the bad practice group because the team would not be able to communicate or meet up. It would be even worse if the team members lived in different time zones, they would then not even be able to chat with each other about the code.

   - Team with members that have different ideas
   This would belong to the bad practice group because everyone will want something else with the project and the members will be working against each other instead of together. This would slow down the project a lot.

4. - Rules & Regulations driven

   In a rules & regulations driven project, the rules and regulations are set from the beginning and there is no space for changes during the project. Because the team will be bound to these rules, they are likely to spend a lot of time on trying to obey them instead of improving and writing new code. Code quality, time and money will be lost in this way. This aspect therefore belongs to the bad practice group.

   - Once-only project

   In a once-only project, everything will have to be done for the first time, and only for this one project. Starting up the project will take quite some time and money, and the skills that the team will learn are only needed for a single time. The once-only project factor belongs to the bad practice group because it will cost the team more time and money than in other same-sized projects.

   - Many team changes, inexperienced team

   When there are team changes, new people in the team have to get familiar with the code first before they can actually start working on the code. This will take some time (and therefore money), especially when there are a lot of team changes. You will end up with an inexperienced team and much time will be spent in understanding and changing code (code quality will likely be not very high either), instead of building towards the end product of the project. Therefore this factor belongs to the bad practice group.