Question Sheet 1

Name: _____ ID: _____

These questions do not have a formal, definitive answer. They are meant to be food for thoughts. Feel free to seek answers on browsing the Internet, talking to other software developers or reading books.

This quiz is about software quality factors. There are many such factors, and each poses a number of questions.

1. (**Correctness**) How to check that a program is correct?

2. (**Robustness**) How the programming language helps on robustness?

3. (**Extensibility**) Could you give examples of a situation that would force the extension of an existing software?

4. (**Reusability**) Think about the similarities between a library check-out system and an address book. What do they have in common?

5. (**Compatibility**) What factors may break compatibility between consecutive releases of the same software?

6. (**Efficiency**) Nowadays we have a tendency of downplaying efficiency in many software development centers. Why this would happen?

7. (**Portability**) Portability became a very strong need on the 90's. Why did that happen?

8. (**Ease of use**) "Easy of use" for one may be a pain to another one. Can you give examples?

9. (**Easy of use**) Why is it important that the developer have a good knowledge about the end user of his product?

10. (**Functionality**) "Simple is better, ..., but sell less". Explain this phrase, in the context of the software industry.

11. (**Timeliness**) This requirement is particularly important in computer science. Why?

12. (**Integrity**) Are there scenarios where safety is not a concern?

13. Many of the factors of software quality might be detrimental of each other. For instance, a robust software will often be slower than a fragile system, due to the robust platform performing more safety checks. Which trade-offs we see in these criteria?

    (a) Integrity $\times$ easy-of-use

    (b) Efficiency $\times$ portability

    (c) timeliness $\times$ everything else

14. Had you given the difficult task of ranking the factors of software quality, which ordering would you choose?

15. Is it possible to measure software quality?