



Curso de Engenharia de Sistemas e Informática - 3º Ano
Engenharia de Software

Ficha T. Prática nº 2

Objectivo:

Reflexão sobre as qualidades e natureza do software e princípios de engenharia de software.

Situação Proposta:

“Os termos correção, fiabilidade e robustez são muitas vezes utilizados de forma permutável para caracterizar a qualidade do software, pretendendo com isso indicar-se que a aplicação executa as suas funções como era esperado”.

Adaptado de “*Fundamentals of Software Engineering*”

1. Comente o excerto anterior.

“A reutilização é aparentada com a capacidade de evolução. Esta última é difícil de conseguir ‘a posteriori’; então, devemos preocupar-nos com ela quando os diversos componentes de software são desenvolvidos”.

Adaptado de “*Fundamentals of Software Engineering*”

2. Comente o excerto acima.

3. Discuta a forma como a reutilização pode afectar a fiabilidade dos produtos de software.

“O software é portátil se pode correr em ambientes diversos. Com a proliferação de processadores e sistemas operativos, a portabilidade tornou-se uma questão importante para os engenheiros de software”.

Adaptado de “*Fundamentals of Software Engineering*”

4. Comente o texto.

5. Discuta a portabilidade como um caso especial da possibilidade de reutilização..

“Para atingir a integração, decomposição e compreensão modular, os módulos devem ter alta coesão e baixo acoplamento. Um módulo tem alta coesão se todos os seus elementos estiverem relacionados de uma forma forte... Enquanto que a coesão é uma propriedade interna dum módulo, o acoplamento caracteriza o relacionamento do módulo com os outros. O acoplamento mede a interdependência de dois módulos. Se dois módulos dependem reciprocamente de forma forte, têm um alto acoplamento.”

Adaptado de “*Fundamentals of Software Engineering*”

6. Porque é desejável que os módulos possuam elevada coesão e um acoplamento fraco?
7. Explique alguma das causas e remédios para a baixa coesão para um módulo de software.
8. Idem para o elevado acoplamento entre dois módulos de software.

“Incrementalidade caracteriza um processo que prossegue sob a forma de degraus sucessivos, em incrementos. A meta desejada é atingida através de aproximações sucessivas, cada vez mais próximas. Cada aproximação é conseguida através de um incremento a partir do anterior...”

Uma forma de aplicar o princípio da Incrementalidade consiste na identificação de subconjuntos iniciais de uma aplicação que possam ser desenvolvidos e entregues ao cliente, por forma a obter um feedback o mais cedo possível. Isto permite que aplicação evolua de forma controlada em casos onde os requisitos iniciais são estáveis ou não estejam completamente compreendidos.”

Adaptado de “*Fundamentals of Software Engineering*”

9. Discuta o relacionamento entre Incrementalidade e oportunidade (entrega do software dentro do prazo).

“Sempre que temos de resolver um problema, deve tentar-se encontrar um problema mais geral que possa estar escondido no problema em mãos. Pode acontecer que o problema genérico não seja mais complexo - pode mesmo ser mais simples do que o problema original. Sendo mais geral, é muito possível que a solução para o problema genérico possa ser reutilizada. Pode mesmo acontecer que a solução possa ser proporcionada por um package disponível no exterior. Pode de igual forma acontecer que ao generalizar-se um problema se acabe por conceber um módulo que seja invocado em mais de um ponto da aplicação, em vez de dispor de várias soluções especializadas.”

Adaptado de “*Fundamentals of Software Engineering*”

10. Discuta o relacionamento entre a generalidade, a antecipação de alterações e a reutilização.