

18. (CESPE – 2004 – ABIN – Analista de Sistema) O modelo de desenvolvimento seqüencial linear, também denominado modelo em cascata, é incompatível com o emprego de técnica de análise orientada a objetos no desenvolvimento de um sistema de informação.

Comentários:

VANTAGENS	DESVANTAGENS
É simples de entender e fácil de aplicar, facilitando o planejamento.	Divisão inflexível do projeto em estágios distintos.
Fixa pontos específicos para a entrega de artefatos.	Dificuldade em incorporar mudanças de requisitos.
Funciona bem para equipes tecnicamente fracas.	Clientes só visualizam resultados próximos ao final do projeto.
É fácil de gerenciar, devido a sua rigidez.	Atrasa a redução de riscos.
Realiza documentação extensa por cada fase ou estágio.	Apenas a fase final produz um artefato de software entregável.
Possibilita boa aderência a outros modelos de processo.	Cliente deve saber todos os requisitos no início do projeto.
Funciona bem com projetos pequenos e com requisitos bem conhecidos.	Não fornece feedback entre as fases.
	Pressupõe que os requisitos ficarão estáveis ao longo do tempo.
	Não funciona bem com projetos complexos e OO, apesar de compatível.

Ele é compatível, mas não é recomendado! *Por que, não?* Imagina um projeto super complexo que utiliza uma análise orientada a objetos (que é um modelo mais sofisticado que a análise estruturada). Lembre-se que, no Modelo em Cascata, você não pode errar, porque se você errar, os riscos de o projeto falhar são enormes! Por essa razão, ele não é recomendável, apesar de compatível!



Gabarito: E

19. (CESPE – 2004 – TRE/AL – Analista de Sistema) O modelo cascata ou ciclo de vida clássico necessita de uma abordagem sistemática, que envolve, em primeiro lugar, o projeto e, em seguida, a análise, a codificação, os testes e a manutenção.

Comentários:

Por Sommerville	Por Royce	Por Pressman (4ª Ed.)	Por Pressman (6ª Ed.)
Análise e Definição de Requisitos	Requisitos de Sistema	Modelagem e Engenharia do Sistema/Informação	Comunicação
Projeto de Sistema e Software	Requisitos de Software	Análise de Requisitos de Software	Planejamento
Implementação e Teste de Unidade	Análise	Projeto	Modelagem
Integração e Teste de Sistema	Projeto	Geração de Código	Construção
Operação e Manutenção	Codificação	Teste e Manutenção	Implantação
	Teste		
	Operação		

Primeiro Projeto e depois Análise? Não, Análise vem antes do Projeto!

Gabarito: E

20. (CESPE – 2008 – MPE/AM – Analista de Sistema) O modelo de desenvolvimento sequencial linear tem como característica principal a produção de uma versão básica, mas funcional, do software desde as primeiras fases.

Comentários:

VANTAGENS

DESVANTAGENS



É simples de entender e fácil de aplicar, facilitando o planejamento.	Divisão inflexível do projeto em estágios distintos.
Fixa pontos específicos para a entrega de artefatos.	Dificuldade em incorporar mudanças de requisitos.
Funciona bem para equipes tecnicamente fracas.	Clientes só visualizam resultados próximos ao final do projeto.
É fácil de gerenciar, devido a sua rigidez.	Atrasa a redução de riscos.
Realiza documentação extensa por cada fase ou estágio.	Apenas a fase final produz um artefato de software entregável.
Possibilita boa aderência a outros modelos de processo.	Cliente deve saber todos os requisitos no início do projeto.
Funciona bem com projetos pequenos e com requisitos bem conhecidos.	Não fornece feedback entre as fases.
	Pressupõe que os requisitos ficarão estáveis ao longo do tempo.
	Não funciona bem com projetos complexos e OO, apesar de compatível.

Pelo contrário, somente nas fases finais que se tem uma versão! Essa definição está mais com cara de modelo de desenvolvimento em prototipagem.

Gabarito: E

21. (VUNESP - 2012 - SPTrans - Analista de Informática) Uma das abordagens do processo de desenvolvimento da engenharia de software prevê a divisão em etapas, em que o fim de uma é a entrada para a próxima. Esse processo é conhecido como modelo:

- a) Transformação.
- b) Incremental.
- c) Evolutivo.
- d) Espiral.
- e) Cascata.

Comentários:



No Modelo em Cascata, *uma fase só se inicia após o término e aprovação da fase anterior*, isto é, há uma sequência de desenvolvimento do projeto. Por exemplo, a Fase 4 só é iniciada após o término e aprovação da Fase 3. A Fase 5 só é iniciada após o término e aprovação da Fase 4. Mas que fases são essas? Bem, agora que complica, porque cada autor resolve criar suas fases!

Conforme vimos em aula, trata-se do Modelo em Cascata.

Gabarito: E

22.(CESGRANRIO – 2010 – PETROBRÁS – Analista de Sistemas – Processos de Negócio) No Ciclo de Vida Clássico, também chamado de Modelo Sequencial Linear ou Modelo Cascata, é apresentada uma abordagem sistemática composta pelas seguintes atividades:

- a) Análise de Requisitos de Software, Projeto, Geração de Código, Teste e Manutenção.
- b) Modelagem e Engenharia do Sistema/Informação, Análise de Requisitos de Software, Projeto, Geração de Código, Teste e Manutenção.
- c) Modelagem e Engenharia do Sistema/Informação, Projeto, Geração de Código, Teste e Manutenção.
- d) Levantamento de Requisitos de Software, Projeto, Geração de Código e Manutenção e Análise de Requisitos de Software.
- e) Levantamento de Requisitos de Software, Projeto, Geração de Código, Teste Progressivo e Manutenção.

Comentários:

Por Sommerville	Por Royce	Por Pressman (4ª Ed.)	Por Pressman (6ª Ed.)
Análise e Definição de Requisitos	Requisitos de Sistema	Modelagem e Engenharia do Sistema/Informação	Comunicação
Projeto de Sistema e Software	Requisitos de Software	Análise de Requisitos de Software	Planejamento

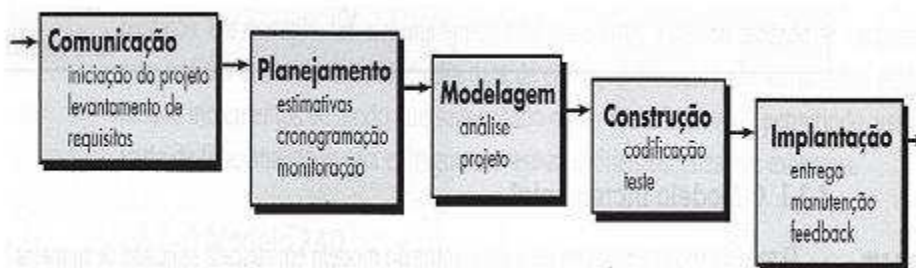


Implementação e Teste de Unidade	Análise	Projeto	Modelagem
Integração e Teste de Sistema	Projeto	Geração de Código	Construção
Operação e Manutenção	Codificação	Teste e Manutenção	Implantação
	Teste		
	Operação		

A Letra B está correta de acordo com o Pressman 4ª Edição, mas está errada de acordo com o Pressman 6ª Edição. Ademais, na questão ele sequer disse que era de acordo com o Pressman. Portanto, percebam que é um assunto polêmico e que as bancas deveriam ignorar, mas eventualmente elas cobram mesmo assim.

Gabarito: B

23. (FGV - 2015 – PGE/RO - Análise de Sistemas) A figura abaixo ilustra um modelo de processo, que prescreve um conjunto de elementos de processo como atividades de arcabouço, ações de engenharia de software, tarefas, produtos de trabalho, mecanismos de garantia de qualidade e de controle de modificações para cada projeto.



Esse modelo é conhecido como Modelo:

- a) por funções.
- b) em cascata.
- c) incremental.
- d) em pacotes.
- e) por módulos.



Comentários:

Por Sommerville	Por Royce	Por Pressman (4ª Ed.)	Por Pressman (6ª Ed.)
Análise e Definição de Requisitos	Requisitos de Sistema	Modelagem e Engenharia do Sistema/Informação	Comunicação
Projeto de Sistema e Software	Requisitos de Software	Análise de Requisitos de Software	Planejamento
Implementação e Teste de Unidade	Análise	Projeto	Modelagem
Integração e Teste de Sistema	Projeto	Geração de Código	Construção
Operação e Manutenção	Codificação	Teste e Manutenção	Implantação
	Teste		
	Operação		

Conforme vimos em aula, trata-se das fases descritas pelo Pressman para o Modelo em Cascata.

Gabarito: B

24.(CESPE - 2016 – TCE/PR - Analista de Informática) O modelo de desenvolvimento em cascata é utilizado em caso de divergência nos requisitos de um software, para permitir a evolução gradual do entendimento dos requisitos durante a implementação do software.

Comentários:

Essa questão trata, na verdade, do modelo iterativo e incremental e, não, em cascata.

Gabarito: E

25.(CESGRANRIO – 2012 – Transpetro – Analista de Sistemas Júnior – Infra-estrutura) Na engenharia de software, existem diversos modelos de



desenvolvimento de software, e, dentre eles, o modelo em cascata, o qual, no contexto do desenvolvimento de sistemas de software complexos, recomenda:

- a) distribuir a elicitación dos requisitos desde o início até o fim do desenvolvimento.
- b) dividir o desenvolvimento do produto de software em fases lineares e sequenciais.
- c) enfatizar a avaliação e mitigação de riscos durante o desenvolvimento.
- d) realizar entregas incrementais do produto de software ao longo do desenvolvimento.
- e) usar prototipagem rápida para estimular o envolvimento do usuário no desenvolvimento.

Comentários:

Recomenda dividir o desenvolvimento do produto de software em fases lineares e sequenciais. *Vocês se lembram que o Modelo em Cascata é um modelo sequencial linear?* Pois é!

Gabarito: B

26. (CESGRANRIO – 2012 – EPE – Analista de Gestão Corporativa – Tecnologia da Informação) Uma das críticas feitas ao modelo do ciclo de vida do desenvolvimento de software em cascata refere-se a:

- a) exigência de conhecimento de avaliação e gerenciamento de risco para evitar grandes surpresas no projeto.
- b) comprometimentos na qualidade e nas possibilidades de manutenção a longo prazo, parecendo um protótipo.
- c) pouca flexibilidade para mudanças futuras, exigindo compromissos nas fases iniciais do projeto.
- d) pouca visibilidade das etapas do processo, tornando cara a documentação de todas as versões dos sistemas.



e) exigências de velocidade as quais levam o engenheiro de software a utilizar linguagens, algoritmos ou ferramentas ineficientes ao longo de todo o projeto.

Comentários:

É conhecida a pouca flexibilidade do Modelo em Cascata em se adaptar a mudanças conforme o projeto progride. Mudanças nas fases iniciais do projeto têm menor impacto do que as que são necessárias nas fase finais do projeto. Por isso, o Modelo em Cascata é usado preferencialmente em projetos com requisitos muito bem definidos e com pouca volatilidade.

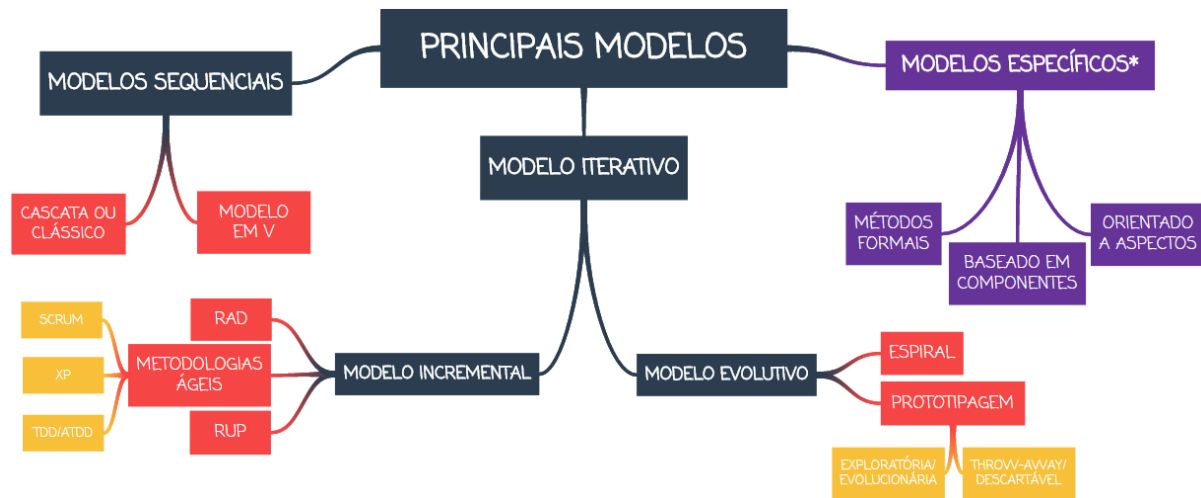
Gabarito: C

ACERTEI	ERREI





MODELOS ITERATIVOS E INCREMENTAIS



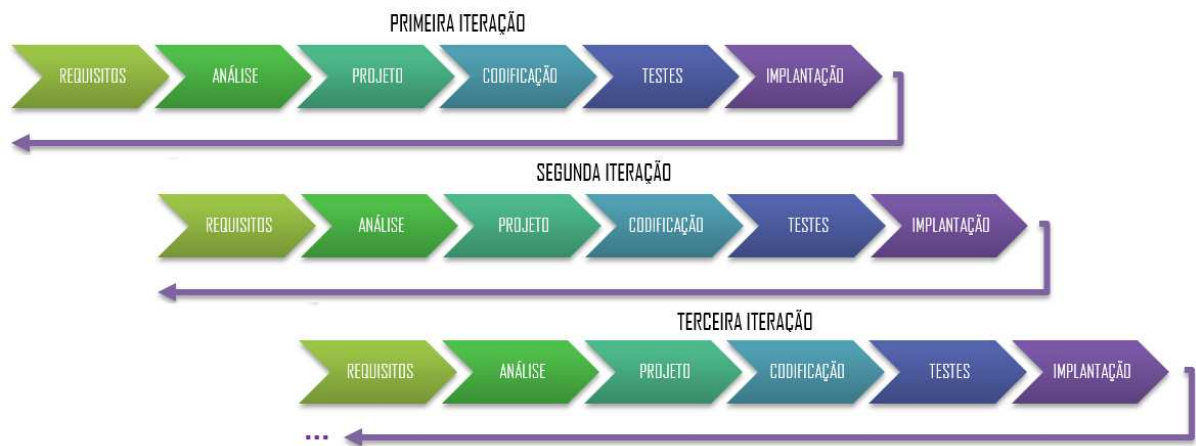
Como foi dito anteriormente, o Modelo em Cascata acumulava riscos e vários projetos começaram a fracassar ao utilizá-lo no mundo inteiro. **Então surgiu o Modelo Iterativo como uma tentativa de resolver esse problema de acúmulo de riscos.** Vejamos a diferença fundamental entre o Modelo em Cascata e o Modelo Iterativo:

- **No Modelo em Cascata**, caso haja cem requisitos, analisam-se os cem requisitos, projetam-se os cem requisitos, codificam-se os cem requisitos, testam-se os cem requisitos, e assim, por diante, sequencialmente.
- **No Modelo Incremental**, caso haja cem requisitos, dividem-se os cem requisitos em vinte miniprojetos de cinco requisitos e utiliza-se o modelo em cascata para cada miniprojeto.
- **No Modelo Iterativo**, caso haja cem requisitos, analisam-se, projetam-se, codificam-se, testam-se os cem requisitos, porém os requisitos são entregues incompletos e eu repito esse ciclo de refinamento até chegar ao produto final.

Galera, pensem só: é possível **combinar a abordagem incremental com uma abordagem iterativa** para desenvolver os miniprojetos e entregar partes diferentes do projeto. A imagem abaixo apresenta miniprojetos sendo feitos iterativamente em um modelo iterativo e incremental. Observem que cada iteração (passagem pelas fases de desenvolvimento) refinam a funcionalidade.

PRIMEIRA ITERAÇÃO





Assim, os resultados são mais rápidos, há maior interação com o usuário e há um feedback mais intenso – é possível reagir mais facilmente a mudanças. **Essa abordagem permite gerenciamento e mitigação de riscos.** Professor, mas eu fiquei com uma dúvida: qual a diferença entre Modelo Iterativo e Modelo Incremental? Ou eles são exatamente a mesma coisa?

Bem, galera... **eu nunca vi nenhuma prova cobrar essa diferença entre modelo iterativo e modelo incremental!** Na verdade, quando se fala em modelo iterativo, presume-se que é incremental (ou evolucionário) e quando se fala em modelo incremental, presume-se que é iterativo. Eles frequentemente andam lado a lado, mas há pequenas diferenças.

IMPORTANTE

Galera, eu já vi essas palavras serem trocadas dezenas de vezes (inclusive em edital). Infelizmente, algumas vezes as bancas não acatam os recursos! De todo modo, saibam que:

- **Iterativo:** reiterado ou repetitivo;
- **Interativo:** participação ou ação mútua.

Professor, mas e se cair em prova? Ora, caso caia em prova, a diferença é que, **no modelo incremental**, há várias equipes desenvolvendo uma parte do software a serem integradas no fim e, **no modelo iterativo**, lança-se a versão 1.0, adicionam-se funcionalidades, lança uma versão 2.0, adicionam-se mais funcionalidades e assim por diante.





Modelo Incremental: observem que a imagem mostra um artista com uma ideia completa sobre o quadro, mas ele desenvolve cada parte separadamente até integrar as partes em uma imagem completa. É como se fosse um quebra-cabeças em que cada parte é entregue funcionando e depois integrada. **Produz builds, i.e., partes do software.**



Modelo Iterativo: observem que a imagem mostra um artista com um esboço do quadro, sendo que ele desenvolve várias versões do quadro até chegar ao resultado final. É como se fosse uma visão abstrata da imagem, que em seguida vai sendo melhorada até chegar a uma visão mais concreta. **Produz releases, i.e., versões constantemente melhoradas da imagem.**

Uma das vantagens do modelo iterativo e incremental é que **o cliente pode receber e avaliar as entregas do produto mais cedo, já no início do desenvolvimento do software**. Além disso, há maior tolerância a mudanças com consequência direta de redução do risco de falha do projeto, i.e., ele acomoda melhor mudanças. Ele aumenta o reúso e a qualidade.



1. (CESPE - 2011 – TJ/ES - Analista Judiciário - Análise de Sistemas - Específicos) O modelo de processo incremental de desenvolvimento de software é iterativo, assim como o processo de prototipagem. Contudo, no processo incremental, diferentemente do que ocorre no de prototipagem, o objetivo consiste em apresentar um produto operacional a cada incremento.

Comentários:

De fato, no modelo iterativo e incremental, apresenta-se sempre um produto a cada incremento. Já na prototipação, não. Idealmente, ele serve apenas para identificar requisitos.

Gabarito: C

2. (CESPE - 2008 – TJ/DF - Analista Judiciário - Análise de Sistemas) No modelo de desenvolvimento incremental, embora haja defasagem entre os períodos de desenvolvimento de cada incremento, os incrementos são desenvolvidos em paralelo.

Comentários:

Questão perfeita. Os incrementos são codificados não exatamente em paralelo – há uma pequena defasagem.

Gabarito: C

3. (CESPE - 2009 – UNIPAMPA - Análise de Sistemas) No modelo de desenvolvimento incremental, a cada iteração são realizadas várias tarefas. Na fase de análise, pode ser feito o refinamento de requisitos e o refinamento do modelo conceitual.

Comentários:



Perfeito, é a fase seguinte à fase de requisitos e busca refiná-los.

Gabarito: C

4. (CESPE - 2016 – TCE/PR – Analista de Sistemas – C) No modelo iterativo de desenvolvimento de software, as atividades são dispostas em estágios sequenciais.

Comentários:

A questão trata do modelo em cascata de desenvolvimento de software. No modelo iterativo e incremental, as atividades são dispostas ao longo de diversas iterações

Gabarito: E

ACERTEI	ERREI



LISTA DE EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

CONCEITOS BÁSICOS DE ENGENHARIA DE SOFTWARE

1. (CESPE - 2013 - TRT - 10ª REGIÃO (DF e TO) - Analista Judiciário - Tecnologia da Informação) A engenharia de software engloba processos, métodos e ferramentas. Um de seus focos é a produção de software de alta qualidade a custos adequados.
2. (FCC - 2012 - TST - Analista Judiciário - Análise de Sistemas) A Engenharia de Software:
 - a) é uma área da computação que visa abordar de modo sistemático as questões técnicas e não técnicas no projeto, implantação, operação e manutenção no desenvolvimento de um software.
 - b) consiste em uma disciplina da computação que aborda assuntos relacionados a técnicas para a otimização de algoritmos e elaboração de ambientes de desenvolvimento.
 - c) trata-se de um ramo da TI que discute os aspectos técnicos e empíricos nos processos de desenvolvimento de sistemas, tal como a definição de artefatos para a modelagem ágil.
 - d) envolve um conjunto de itens que abordam os aspectos de análise de mercado, concepção e projeto de software, sendo independente da engenharia de um sistema.
 - e) agrupa as melhores práticas para o concepção, projeto, operação e manutenção de artefatos que suportam a execução de programas de computador, tais como as técnicas de armazenamento e as estruturas em memória principal.
3. (FCC - 2012 - TRT - 6ª Região (PE) - Técnico Judiciário - Tecnologia da Informação) Considere: *é uma disciplina que se ocupa de todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até a manutenção desse sistema, depois que ele entrou em operação. Seu principal objetivo é fornecer uma estrutura metodológica para a construção de software com alta qualidade.* A definição refere-se:



- a) ao ciclo de vida do software.
- b) à programação orientada a objetos.
- c) à análise de sistemas.
- d) à engenharia de requisitos.
- e) à engenharia de software.

4. (CESPE - 2011 - MEC - Gerente de Projetos) A engenharia de software, disciplina relacionada aos aspectos da produção de software, abrange somente os processos técnicos do desenvolvimento de software.
5. (FGV - 2010 - BADESC - Analista de Sistemas - Desenvolvimento de Sistemas) De acordo com Pressman, a engenharia de software é baseada em camadas, com foco na qualidade.

Essas camadas são:

- a) métodos, processo e teste.
 - b) ferramentas, métodos e processo.
 - c) métodos, construção, teste e implantação.
 - d) planejamento, modelagem, construção, validação e implantação.
 - e) comunicação, planejamento, modelagem, construção e implantação.
6. (CESPE - 2010 - Banco da Amazônia - Técnico Científico - Tecnologia da Informação) Os princípios de engenharia de software definem a necessidade de formalidades para reduzir inconsistências e a decomposição para lidar com a complexidade.
7. (FCC - 2010 - TRE-AM - Analista Judiciário - Tecnologia da Informação - A) A Engenharia de Software:
- a) não tem como método a abordagem estruturada para o desenvolvimento de software, pois baseia-se exclusivamente nos modelos de software, notações, regras e técnicas de desenvolvimento.
 - b) se confunde com a Ciência da Computação quando ambas tratam do desenvolvimento de teorias, fundamentações e práticas de desenvolvimento de software.



c) tendo como foco apenas o tratamento dos aspectos de construção de software, subsidia a Engenharia de Sistemas no tratamento dos sistemas baseados em computadores, incluindo hardware e software.

d) tem como foco principal estabelecer uma abordagem sistemática de desenvolvimento, através de ferramentas e técnicas apropriadas, dependendo do problema a ser abordado, considerando restrições e recursos disponíveis.

e) segue princípios, tais como, o da Abstração, que identifica os aspectos importantes sem ignorar os detalhes e o da Composição, que agrupa as atividades em um único processo para distribuição aos especialistas.

8. (FCC - 2011 - INFRAERO - Analista de Sistemas - Gestão de TI) Em relação à Engenharia de Software, é INCORRETO afirmar:

a) O design de software, ao descrever os diversos aspectos que estarão presentes no sistema quando construído, permite que se faça a avaliação prévia para garantir que ele alcance os objetivos propostos pelos interessados.

b) A representação de um design de software mais simples para representar apenas as suas características essenciais busca atender ao princípio da abstração.

c) Iniciar a entrevista para obtenção dos requisitos de software com perguntas mais genéricas e finalizar com perguntas mais específicas sobre o sistema é o que caracteriza a técnica de entrevista estruturada em funil.

d) No contexto de levantamento de requisitos, funcionalidade é um dos aspectos que deve ser levado em conta na abordagem dos requisitos funcionais.

e) A representação é a linguagem do design, cujo único propósito é descrever um sistema de software que seja possível construir.

9. (FCC – 2009 – AFR/SP - Analista de Sistemas) A engenharia de software está inserida no contexto:

a) das engenharias de sistemas, de processo e de produto.

b) da engenharia de sistemas, apenas.

c) das engenharias de processo e de produto, apenas.

d) das engenharias de sistemas e de processo, apenas.

e) das engenharias de sistemas e de produto, apenas.



10. (CESPE – 2015 – STJ – Analista de Sistemas) Embora os engenheiros de software geralmente utilizem uma abordagem sistemática, a abordagem criativa e menos formal pode ser eficiente em algumas circunstâncias, como, por exemplo, para o desenvolvimento de sistemas web, que requerem uma mistura de habilidades de software e de projeto.
11. (CESPE – 2015 – STJ – Analista de Sistemas) O foco da engenharia de software inclui especificação do sistema, desenvolvimento de hardware, elaboração do projeto de componentes de hardware e software, definição dos processos e implantação do sistema.
12. (FUNIVERSA – 2009 – IPHAN – Analista de Sistemas) A Engenharia de Software resume-se em um conjunto de técnicas utilizadas para o desenvolvimento e manutenção de sistemas computadorizados, visando produzir e manter softwares de forma padronizada e com qualidade. Ela obedece a alguns princípios como (1) Formalidade, (2) Abstração, (3) Decomposição, (4) Generalização e (5) Flexibilização. Assinale a alternativa que apresenta conceito correto sobre os princípios da Engenharia de Software.
- a) A flexibilização é o processo que permite que o software possa ser alterado, sem causar problemas para sua execução.
- b) A formalidade é a maneira usada para resolver um problema, de forma genérica, com o intuito de poder reaproveitar essa solução em outras situações semelhantes.
- c) A generalização preocupa-se com a identificação de um determinado fenômeno da realidade, sem se preocupar com detalhes, considerando apenas os aspectos mais relevantes.
- d) Pelo princípio da decomposição, o software deve ser desenvolvido de acordo com passos definidos com precisão e seguidos de maneira efetiva.
- e) A abstração é a técnica de se dividir o problema em partes, de maneira que cada uma possa ser resolvida de uma forma mais específica.
13. (CESPE – 2013 – TCE/RO – Analista de Sistemas) Engenharia de software não está relacionada somente aos processos técnicos de desenvolvimento de softwares,



mas também a atividades como gerenciamento de projeto e desenvolvimento de ferramentas, métodos e teorias que apoiem a produção de softwares.

14. (CESPE – 2010 – DETRAN/ES – Analista de Sistemas) Segundo princípio da engenharia de software, os vários artefatos produzidos ao longo do seu ciclo de vida apresentam, de forma geral, nível de abstração cada vez menor.
15. (CESPE – 2010 – TRE/BA – Analista de Sistemas) Entre os desafios enfrentados pela engenharia de software estão lidar com sistemas legados, atender à crescente diversidade e atender às exigências quanto a prazos de entrega reduzidos.
16. (FCC – 2010 – DPE/SP – Analista de Sistemas) A Engenharia de Software

I. não visa o desenvolvimento de teorias e fundamentações, preocupando-se unicamente com as práticas de desenvolvimento de software.

II. tem como foco o tratamento dos aspectos de desenvolvimento de software, abstraindo-se dos sistemas baseados em computadores, incluindo hardware e software.

III. tem como métodos as abordagens estruturadas para o desenvolvimento de software que incluem os modelos de software, notações, regras e maneiras de desenvolvimento.

IV. segue princípios, tais como, o da Abstração, que identifica os aspectos importantes sem ignorar os detalhes e o da Composição, que agrupa as atividades em um único processo para distribuição aos especialistas.

É correto o que se afirma em:

- a) III e IV, apenas.
 - b) I, II, III e IV.
 - c) I e II, apenas.
 - d) I, II e III, apenas.
 - e) II, III e IV, apenas.
17. (CESPE – 2013 – TCE/RO – Analista de Sistemas) Assim como a Engenharia de Software, existe também na área de informática a chamada Ciência da



Computação. Assinale a alternativa que melhor apresenta a diferença entre Engenharia de Software e Ciência da Computação.

- a) A Ciência da Computação tem como objetivo o desenvolvimento de teorias e fundamentações. Já a Engenharia de Software se preocupa com as práticas de desenvolvimento de software.
 - b) A Engenharia de Software trata da criação dos sistemas de computação (softwares) enquanto a Ciência da Computação está ligada ao desenvolvimento e criação de componentes de hardware.
 - c) A Engenharia de Software trata dos sistemas com base em computadores, que inclui hardware e software, e a Ciência da Computação trata apenas dos aspectos de desenvolvimento de sistemas.
 - d) A Ciência da Computação trata dos sistemas com base em computadores, que inclui hardware e software, e a Engenharia de Software trata apenas dos aspectos de desenvolvimento de sistemas.
 - e) A Ciência da Computação destina-se ao estudo e solução para problemas genéricos das áreas de rede e banco de dados e a Engenharia de Software restringe-se ao desenvolvimento de sistemas.
18. (CESPE – 2009 – ANAC – Analista de Sistemas) O termo engenharia pretende indicar que o desenvolvimento de software submete-se a leis similares às que governam a manufatura de produtos industriais em engenharias tradicionais, pois ambos são metodológicos.
19. (CESPE - 2016 – TCE/PR – Analista de Sistemas – B) A engenharia de software refere-se ao estudo das teorias e fundamentos da computação, ficando o desenvolvimento de software a cargo da ciência da computação.
20. (CESPE - 2016 – TCE/PR – Analista de Sistemas – E) O conceito de software se restringe ao desenvolvimento do código em determinada linguagem e seu armazenamento em arquivos.



LISTA DE EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

CICLO DE VIDA E PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE

1. (CESPE – 2009 – TCE/TO – Analista de Sistemas - A) Quanto maior e mais complexo o projeto de software, mais simples deve ser o modelo de processo a ser adotado.
2. (CESPE – 2009 – TCE/TO – Analista de Sistemas - B) O modelo de ciclo de vida do software serve para delimitar o alvo do software. Nessa visão, não são consideradas as atividades necessárias e o relacionamento entre elas.
3. (CESPE – 2009 – TCE/TO – Analista de Sistemas - C) A escolha do modelo do ciclo de vida não depende de características específicas do projeto, pois o melhor modelo é sempre o mais usado pela equipe do projeto.
4. (ESAF - 2012 – CGU – Analista de Sistemas) A escolha de um modelo é fortemente dependente das características do projeto. Os principais modelos de ciclo de vida podem ser agrupados em três categorias principais:
 - a) sequenciais, cascata e evolutivos.
 - b) sequenciais, incrementais e ágeis.
 - c) sequenciais, incrementais e evolutivos.
 - d) sequenciais, ágeis e cascata.
 - e) cascata, ágeis e evolutivos.
5. (CESPE – 2011 – MEC – Analista de Sistemas) O processo de desenvolvimento de software é uma caracterização descritiva ou prescritiva de como um produto de software deve ser desenvolvido.
6. (CESPE – 2013 – TRT/10ª – Analista de Sistemas) As atividades fundamentais relacionadas ao processo de construção de um software incluem a especificação, o desenvolvimento, a validação e a evolução do software.
7. (CESPE – 2010 – TRE/BA – Analista de Sistemas) Um modelo de processo de software consiste em uma representação complexa de um processo de software, apresentada a partir de uma perspectiva genérica.



8. (CESPE – 2011 – MEC – Analista de Sistemas) Atividades comuns a todos os processos de software incluem a especificação, o projeto, a implementação e a validação.
9. (CESPE – 2015 – STJ – Analista de Sistemas) As principais atividades de engenharia de software são especificação, desenvolvimento, validação e evolução.
10. (FCC – 2012 – MPE/AP – Analista de Sistemas) Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software. Existem muitos processos de software diferentes, mas todos devem incluir quatro atividades fundamentais: especificação, projeto e implementação, validação e:
 - a) teste
 - b) evolução.
 - c) prototipação.
 - d) entrega.
 - e) modelagem.
11. (CESPE – 2010 – EMBASA – Analista de Sistemas) Ciclo de vida de um software resume-se em eventos utilizados para definir o status de um projeto.
12. (CESPE – 2016 – TCE/PR – Analista de Sistemas) As fases do ciclo de vida de um software são:
 - a) concepção, desenvolvimento, entrega e encerramento.
 - b) iniciação, elaboração, construção e manutenção.
 - c) escopo, estimativas, projeto e processo e gerência de riscos.
 - d) análise, desenvolvimento, teste, empacotamento e entrega.
 - e) planejamento, análise e especificação de requisitos, projeto, implementação, testes, entrega e implantação, operação e manutenção.
13. (MPE/RS – 2012 – MPE/RS – Analista de Sistemas) O ciclo de vida básico de um software compreende:
 - a) a implementação, a implantação e o teste.
 - b) a análise, a segurança e o controle de usuários.
 - c) a implementação, a análise e o teste.
 - d) a implementação, a validação e as vendas.



e) a análise, o projeto, a implementação e o teste.

14. (CESGRANRIO – 2011 – PETROBRÁS – Analista de Sistemas) A especificação de uma Metodologia de Desenvolvimento de Sistemas tem como pré-requisito indispensável, em relação ao que será adotado no processo de desenvolvimento, a definição do:
- a) Engenheiro Responsável pelo Projeto
 - b) Documento de Controle de Sistemas
 - c) Software para Desenvolvimento
 - d) Ciclo de Vida do Software
 - e) Bloco de Atividades
15. (CESPE – 2008 – INPE – Analista de Sistemas) O ciclo de vida do software tem início na fase de projeto.
16. (CESPE – 2013 – TRT/10 – Analista de Sistemas) O ciclo de vida de um software, entre outras características, está relacionado aos estágios de concepção, projeto, criação e implementação.
17. (CESPE – 2011 – TJ/ES – Analista de Sistemas) Entre as etapas do ciclo de vida de software, as menos importantes incluem a garantia da qualidade, o projeto e o estudo de viabilidade. As demais atividades do ciclo, como a implementação e os testes, requerem maior dedicação da equipe e são essenciais.
18. (CESPE - 2016 – TCE/PR – Analista de Sistemas – A) A engenharia de software está relacionada aos diversos aspectos de produção de software e inclui as atividades de especificação, desenvolvimento, validação e evolução de software.
19. (CESPE - 2016 – TCE/PR – Analista de Sistemas – D) Um processo de software é composto por quatro atividades fundamentais: iniciação, desenvolvimento, entrega e encerramento.
20. (CESPE - 2016 – TCE/PR – Analista de Sistemas – B) A metodologia de processos genérica para a engenharia de software é composta de seis etapas: comunicação, planejamento, modelagem, construção, emprego e aceitação.
21. (INSTITUTO CIDADE – 2012 – TCM/GO – Analista de Sistemas) De acordo com a engenharia de software, como todo produto industrial, o software possui um ciclo de vida. Cada fase do ciclo de vida possui divisões e subdivisões. Em qual



fase avaliamos a necessidade de evolução dos softwares em funcionamento para novas plataformas operacionais ou para a incorporação de novos requisitos?

- a) Fase de operação;
- b) Fase de retirada;
- c) Fase de definição;
- d) Fase de design.
- e) Fase de desenvolvimento;

22. (CESPE - 2010 – DETRAN/ES – Analista de Sistemas) Quando um aplicativo de software desenvolvido em uma organização atinge, no fim do seu ciclo de vida, a fase denominada aposentadoria, descontinuação ou fim de vida, todos os dados por ele manipulados podem ser descartados.

LISTA DE EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

MODELO EM CASCATA

1. (FCC - 2012 - TJ-RJ - Analista Judiciário - Análise de Sistemas - E) Dos diferentes modelos para o ciclo de vida de desenvolvimento de um software é correto afirmar que o modelo em cascata é o mais recente e complexo.
2. (FCC - 2009 - SEFAZ-SP - Agente Fiscal de Rendas - Tecnologia da Informação - Prova 3 - B) O processo de engenharia de software denominado ciclo de vida clássico refere-se ao modelo incremental.
3. (CESPE – 2009 – INMETRO – Analista de Sistemas) Em uma empresa que tenha adotado um processo de desenvolvimento de software em cascata, falhas no levantamento de requisitos têm maior possibilidade de gerar grandes prejuízos do que naquelas que tenham adotado desenvolvimento evolucionário.
4. (CESPE – 2011 – MEC – Analista de Sistemas) O modelo Waterfall tem a vantagem de facilitar a realização de mudanças sem a necessidade de retrabalho em fases já completadas.
5. (CESPE – 2008 – TST – Analista de Sistemas) No modelo de desenvolvimento sequencial linear, a fase de codificação é a que gera erros de maior custo de correção.



6. (CESPE – 2009 – INMETRO – Analista de Sistemas) Em um processo de desenvolvimento em cascata, os testes de software são realizados todos em um mesmo estágio, que acontece após a finalização das fases de implementação.
7. (CESPE – 2008 – SERPRO – Analista de Sistemas) O modelo em cascata consiste de fases e atividades que devem ser realizadas em sequência, de forma que uma atividade é requisito da outra.
8. (CESPE – 2005 – AL/ES – Analista de Sistemas - B) O modelo de desenvolvimento em cascata descreve ciclos sequenciais, incrementais e iterativos, possuindo, entre outras, as fases de requisitos e implementação.
9. (CESPE – 2004 – STJ – Analista de Sistemas) O modelo de desenvolvimento sequencial linear, também chamado modelo clássico ou modelo em cascata, caracteriza-se por não acomodar adequadamente as incertezas que existem no início de um projeto de software, em especial as geradas pela dificuldade do cliente de explicitar todos os requerimentos que o programa deve contemplar.
10. (CESPE – 2009 – IPEA – Analista de Sistema) No modelo em cascata de processo de desenvolvimento, os clientes devem definir os requisitos apenas durante a fase de projeto; e os projetistas definem as estratégias de projeto apenas durante a fase de implementação. As fases do ciclo de vida envolvem definição de requisitos, projeto, implementação, teste, integração, operação e manutenção. Em cada fase do ciclo de vida, podem ser produzidos diversos artefatos.
11. (CESPE – 2008 – TCE/TO – Analista de Sistema – D) No ciclo de vida em cascata, é possível realizar alternadamente e simultaneamente as atividades de desenvolvimento de software.
12. (CESPE – 2004 – TJ/PA – Analista de Sistema – D) A abordagem sistemática estritamente linear para o desenvolvimento de software é denominada modelo em cascata ou modelo sequencial linear.
13. (CESPE – 2006 – TSE – Analista de Sistema – D) O modelo em cascata organiza o desenvolvimento em fases. Esse modelo encoraja a definição dos requisitos antes do restante do desenvolvimento do sistema. Após a especificação e a análise dos requisitos, têm-se o projeto, a implementação e o teste.
14. (CESPE – 2009 – INMTRO – Analista de Sistema) No desenvolvimento de software, o modelo em cascata é estruturado de tal maneira que as fases que



compõem o desenvolvimento são interligadas. Nessa situação, o final de uma fase implica o início de outra.

15. (CESPE – 2010 – BASA – Analista de Sistema) No modelo em cascata, o projeto segue uma série de passos ordenados. Ao final de cada projeto, a equipe de projeto finaliza uma revisão. O desenvolvimento continua e, ao final, o cliente avalia a solução proposta.
16. (CESPE – 2009 – TRE/MT – Analista de Sistema - A) O modelo em cascata é apropriado para software em que os requisitos ainda não foram bem compreendidos, pois é focado na criação de incrementos.
17. (CESPE – 2009 – UNIPAMPA – Analista de Sistema – D) O modelo em cascata sugere uma abordagem sistemática e sequencial para o desenvolvimento de software. Sua natureza linear leva a estados de bloqueio nos quais, para que nova etapa seja iniciada, é necessário que a documentação associada à fase anterior tenha sido aprovada.
18. (CESPE – 2004 – ABIN – Analista de Sistema) O modelo de desenvolvimento sequencial linear, também denominado modelo em cascata, é incompatível com o emprego de técnica de análise orientada a objetos no desenvolvimento de um sistema de informação.
19. (CESPE – 2004 – TRE/AL – Analista de Sistema) O modelo cascata ou ciclo de vida clássico necessita de uma abordagem sistemática, que envolve, em primeiro lugar, o projeto e, em seguida, a análise, a codificação, os testes e a manutenção.
20. (CESPE – 2008 – MPE/AM – Analista de Sistema) O modelo de desenvolvimento sequencial linear tem como característica principal a produção de uma versão básica, mas funcional, do software desde as primeiras fases.
21. (VUNESP - 2012 - SPTrans - Analista de Informática) Uma das abordagens do processo de desenvolvimento da engenharia de software prevê a divisão em etapas, em que o fim de uma é a entrada para a próxima. Esse processo é conhecido como modelo:
 - a) Transformação.
 - b) Incremental.
 - c) Evolutivo.
 - d) Espiral.



e) Cascata.

22. (CESGRANRIO – 2010 – PETROBRÁS – Analista de Sistemas – Processos de Negócio) No Ciclo de Vida Clássico, também chamado de Modelo Sequencial Linear ou Modelo Cascata, é apresentada uma abordagem sistemática composta pelas seguintes atividades:

a) Análise de Requisitos de Software, Projeto, Geração de Código, Teste e Manutenção.

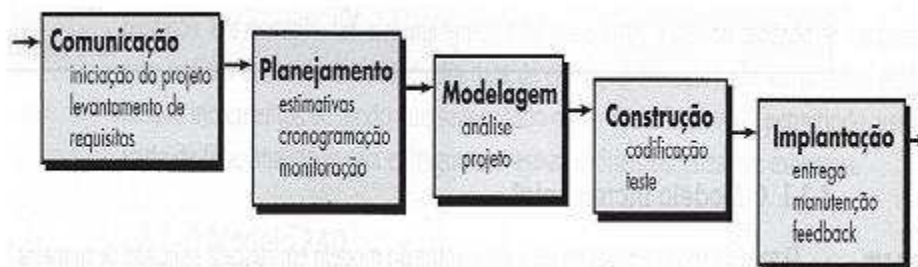
b) Modelagem e Engenharia do Sistema/Informação, Análise de Requisitos de Software, Projeto, Geração de Código, Teste e Manutenção.

c) Modelagem e Engenharia do Sistema/Informação, Projeto, Geração de Código, Teste e Manutenção.

d) Levantamento de Requisitos de Software, Projeto, Geração de Código e Manutenção e Análise de Requisitos de Software.

e) Levantamento de Requisitos de Software, Projeto, Geração de Código, Teste Progressivo e Manutenção.

23. (FGV - 2015 – PGE/RO - Análise de Sistemas) A figura abaixo ilustra um modelo de processo, que prescreve um conjunto de elementos de processo como atividades de arcabouço, ações de engenharia de software, tarefas, produtos de trabalho, mecanismos de garantia de qualidade e de controle de modificações para cada projeto.



Esse modelo é conhecido como Modelo:

a) por funções.

b) em cascata.



- c) incremental.
- d) em pacotes.
- e) por módulos.

24. (CESPE - 2016 – TCE/PR - Analista de Informática) O modelo de desenvolvimento em cascata é utilizado em caso de divergência nos requisitos de um software, para permitir a evolução gradual do entendimento dos requisitos durante a implementação do software.

25. (CESGRANRIO – 2012 – Transpetro – Analista de Sistemas Júnior – Infra-estrutura) Na engenharia de software, existem diversos modelos de desenvolvimento de software, e, dentre eles, o modelo em cascata, o qual, no contexto do desenvolvimento de sistemas de software complexos, recomenda:

- a) distribuir a elicitação dos requisitos desde o início até o fim do desenvolvimento.
- b) dividir o desenvolvimento do produto de software em fases lineares e sequenciais.
- c) enfatizar a avaliação e mitigação de riscos durante o desenvolvimento.
- d) realizar entregas incrementais do produto de software ao longo do desenvolvimento.
- e) usar prototipagem rápida para estimular o envolvimento do usuário no desenvolvimento.

26. (CESGRANRIO – 2012 – EPE – Analista de Gestão Corporativa – Tecnologia da Informação) Uma das críticas feitas ao modelo do ciclo de vida do desenvolvimento de software em cascata refere-se a:

- a) exigência de conhecimento de avaliação e gerenciamento de risco para evitar grandes surpresas no projeto.
- b) comprometimentos na qualidade e nas possibilidades de manutenção a longo prazo, parecendo um protótipo.
- c) pouca flexibilidade para mudanças futuras, exigindo compromissos nas fases iniciais do projeto.



d) pouca visibilidade das etapas do processo, tornando cara a documentação de todas as versões dos sistemas.

e) exigências de velocidade as quais levam o engenheiro de software a utilizar linguagens, algoritmos ou ferramentas ineficientes ao longo de todo o projeto.

LISTA DE EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

MODELOS ITERATIVOS E INCREMENTAIS

1. (CESPE - 2011 – TJ/ES - Analista Judiciário - Análise de Sistemas - Específicos) O modelo de processo incremental de desenvolvimento de software é iterativo, assim como o processo de prototipagem. Contudo, no processo incremental, diferentemente do que ocorre no de prototipagem, o objetivo consiste em apresentar um produto operacional a cada incremento.
2. (CESPE - 2008 – TJ/DF - Analista Judiciário - Análise de Sistemas) No modelo de desenvolvimento incremental, embora haja defasagem entre os períodos de desenvolvimento de cada incremento, os incrementos são desenvolvidos em paralelo.
3. (CESPE - 2009 – UNIPAMPA - Análise de Sistemas) No modelo de desenvolvimento incremental, a cada iteração são realizadas várias tarefas. Na fase de análise, pode ser feito o refinamento de requisitos e o refinamento do modelo conceitual.
4. (CESPE - 2016 – TCE/PR – Analista de Sistemas – C) No modelo iterativo de desenvolvimento de software, as atividades são dispostas em estágios sequenciais.



GABARITO DOS EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

CONCEITOS BÁSICOS DE ENGENHARIA DE SOFTWARE

1	2	3	4	5	6	7	8	9	10
C	A	E	E	B	C	D	E	A	C
11	12	13	14	15	16	17	18	19	20
E	A	C	C	C	D	A	C	E	E

GABARITO DOS EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

CICLO DE VIDA E PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE

1	2	3	4	5	6	7	8	9	10
E	E	E	C	C	C	E	C	C	B
11	12	13	14	15	16	17	18	19	20
E	E	E	D	E	C	E	C	E	E
21	22	23	24	25	26	27	28	29	30
B	E								

GABARITO DOS EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

MODELO EM CASCATA

1	2	3	4	5	6	7	8	9	10
E	E	C	E	E	E	C	E	C	E
11	12	13	14	15	16	17	18	19	20
E	C	C	C	E	E	C	E	E	E
21	22	23	24	25	26	27	28	29	30
E	B	B	E	B	C				

GABARITO DOS EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

MODELOS ITERATIVOS E INCREMENTAIS

1	2	3	4	5	6	7	8	9	10
C	C	C	E						



ESSA LEI TODO MUNDO CONHECE: PIRATARIA É CRIME.

Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.



1 Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.



2 Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).



3 Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).



4 Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).



5 Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).



6 Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).



7 Concurseiro(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.



8 O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.