

CÂMARA LEGISLATIVA DO DISTRITO FEDERAL

Nome do candidato:

Número do documento de identidade:

Número de inscrição:

Sala:

Sequencial:

Concurso Público



Aplicação: 22/1/2006

Cargo 13:

TARDE

Consultor Técnico Legislativo

Categoria: **Analista de Sistemas – Área 2**

LEIA COM ATENÇÃO AS INSTRUÇÕES ABAIXO.

- 1 Ao receber este caderno, confira inicialmente os seus dados pessoais transcritos acima. Em seguida, verifique se ele contém cento e vinte itens, correspondentes às provas objetivas, corretamente ordenados de 1 a 120, seguidos da prova discursiva.
- 2 Caso os dados pessoais constantes neste caderno não correspondam aos seus, ou, ainda, caso o caderno esteja incompleto ou tenha qualquer defeito, solicite ao fiscal de sala mais próximo que tome as providências cabíveis.
- 3 O espaço para rascunho da prova discursiva é de uso opcional; não contará, portanto, para efeito de avaliação.
- 4 Nos itens das provas objetivas, recomenda-se não marcar ao acaso: em cada item, se a resposta divergir do gabarito oficial definitivo, o candidato receberá pontuação negativa, conforme consta em edital.
- 5 Não utilize lápis, lapiseira, borracha e(ou) qualquer material de consulta que não seja fornecido pelo CESPE.
- 6 Não serão distribuídas folhas suplementares para rascunho nem para texto definitivo.
- 7 Durante as provas, não se comunique com outros candidatos nem se levante sem autorização do chefe de sala.
- 8 A duração das provas é de **quatro horas e trinta minutos**, já incluído o tempo destinado à identificação — que será feita no decorrer das provas —, ao preenchimento da folha de respostas e à transcrição do texto definitivo para a respectiva folha.
- 9 Na prova discursiva, não será avaliado texto escrito a lápis, texto escrito em local indevido ou texto que tenha identificação fora do local apropriado.
- 10 Ao terminar as provas, chame o fiscal de sala mais próximo, devolva-lhe a sua folha de respostas e a folha de texto definitivo da prova discursiva e deixe o local de provas.
- 11 A desobediência a qualquer uma das determinações constantes no presente caderno, na folha de rascunho, na folha de respostas ou na folha de texto definitivo da prova discursiva poderá implicar a anulação das suas provas.

AGENDA

- I 24/1/2006, a partir das 17 h (horário de Brasília) – Gabaritos oficiais preliminares das provas objetivas: Internet — www.cespe.unb.br/concursos/camaradf2005.
- II 25 a 27/1/2006 – Recursos (provas objetivas): exclusivamente no Sistema Eletrônico de Interposição de Recurso, Internet — www.cespe.unb.br/concursos/camaradf2005, mediante instruções e formulários que estarão disponíveis nesse endereço.
- III 14/2/2006 – Resultado final das provas objetivas e resultado provisório da prova discursiva: Diário Oficial do Distrito Federal, Diário da Câmara Legislativa e Internet — www.cespe.unb.br/concursos/camaradf2005.
- IV 15 a 21/2/2006 – Recursos (prova discursiva): em locais e horários que serão informados na divulgação do resultado provisório.
- V 14/3/2006 – Resultado final da prova discursiva e convocação para a entrega de títulos: locais mencionados no item III.

OBSERVAÇÕES

- Não serão objeto de conhecimento recursos em desacordo com o item 24 do Edital n.º 1/2005 – CLDF, de 26/10/2005.
- Informações adicionais: telefone 0(XX) 61 3448 0100; Internet — www.cespe.unb.br.
- É permitida a reprodução deste material apenas para fins didáticos, desde que citada a fonte.

UnBC **ESPE**
UNIVERSIDADE DE BRASÍLIA

- De acordo com o comando a que cada um dos itens de **1 a 120** se refira, marque, na **folha de respostas**, para cada item: o campo designado com o código **C**, caso julgue o item **CERTO**; ou o campo designado com o código **E**, caso julgue o item **ERRADO**. A ausência de marcação ou a marcação de ambos os campos não serão apenadas, ou seja, não receberão pontuação negativa. Para as devidas marcações, use a folha de rascunho e, posteriormente, a **folha de respostas**, único documento válido para a correção das suas provas.
- As siglas seguintes, quando usadas, deverão ser interpretadas da forma indicada: DF = Distrito Federal; LODF = Lei Orgânica do Distrito Federal; CLDF = Câmara Legislativa do Distrito Federal; TJDFT = Tribunal de Justiça do Distrito Federal e Territórios; STJ = Superior Tribunal de Justiça; STF = Supremo Tribunal Federal.

CONHECIMENTOS BÁSICOS

1 O Brasil é grande. Apesar de todas as crueldades e
discriminações, especialmente contra as comunidades
indígenas e negras, e de todas as desigualdades e dores que
4 não devemos esquecer jamais, o povo brasileiro realizou
uma obra de resistência e construção nacional admirável.
Construiu, ao longo do século, uma nação plural,
7 diversificada, contraditória até, mas que se estende de uma
ponta a outra do território. Dos encantados da Amazônia aos
orixás da Bahia; do frevo pernambucano às escolas de samba
10 do Rio de Janeiro; dos tambores do Maranhão ao barroco
mineiro; da arquitetura de Brasília à música sertaneja.
Estendendo o arco de sua multiplicidade nas culturas de São
13 Paulo, do Paraná, de Santa Catarina, do Rio Grande do Sul
e da região Centro-Oeste. Esta é uma nação que fala a
mesma língua, partilha os mesmos valores fundamentais,
16 sente-se brasileira.

Onde a mestiçagem e o sincretismo se impuseram,
dando uma contribuição original ao mundo, onde judeus e
19 árabes conversam sem medo, onde, toda migração é bem-
vinda, porque sabemos que, em pouco tempo, pela nossa
própria capacidade de assimilação e de bem-querer, cada
22 migrante se transforma em mais um brasileiro.

Luis Inácio Lula da Silva. *Discurso de Posse.*

Com referência aos sentidos e às estruturas lingüísticas do texto
acima, julgue os itens seguintes.

- 1 Os sinais de ponto-e-vírgula às linhas de 9 a 11 podem, sem prejuízo para a correção gramatical do período, ser substituídos por vírgulas.
- 2 A expressão “é grande” (ℓ.1) restringe-se à extensão territorial do Brasil.
- 3 A inserção de **Essa nação vai dos** no lugar de “Dos”, antes de “encantados da Amazônia” (ℓ.8), prejudica a correção gramatical do período e altera o significado original da informação.
- 4 Antes de “Onde a mestiçagem” (ℓ.17), subentende-se a repetição da expressão “Esta é uma nação”, que inicia o período antecedente.

1 O Estado democrático estabelece o direito, a fim de
que o limite da liberdade de cada um seja a liberdade dos
outros. O Brasil livrou-se da tutela do arbítrio e não aceita a
4 tutela da coação, nem o intimidam facções ou grupos.

A liberdade implica o compromisso de fortalecer o
poder político contra a insegurança de abalos institucionais.
7 Ouvir a todos e conviver com todos, sem discriminação.
Tolerância não significa concordância(...)

A liberdade não se esgota na vontade institucional.
10 Ela tem de ser capaz de gerar direitos sociais, para ser a
liberdade que não permita a morte pela fome, pelas doenças,
pela insegurança das cidades e pela ausência de trabalho.
13 Enfim, a liberdade é a vida; é uma perspectiva de vida feliz.

Mas o exercício da liberdade tem de ser integral.
Indissociáveis são as liberdades política, econômica e social.
16 Todos sabem que, onde morreu a liberdade econômica ou
existe a servidão social, a liberdade política não existe.
Querer a liberdade política sem garantir o poder criador
19 competitivo da iniciativa privada é não conhecer a realidade
da História. Querer liberdade econômica, convivendo com a
injustiça social e com a miséria, é admitir uma sociedade de
22 privilégios que termina na violência e no silêncio das
ideologias.

José Sarney. *Discurso de Posse.*

Em relação ao texto acima, julgue os itens a seguir.

- 5 O trecho “ser integral. Indissociáveis são as liberdades” (ℓ.14-15) admite, sem prejuízo para a correção gramatical do texto, a seguinte reescrita: ser integral, uma vez que são indissociáveis as liberdades.
- 6 A expressão “a fim de que” (ℓ.1-2) pode, sem prejuízo para a correção gramatical e para o sentido original do período, ser substituída por qualquer uma das seguintes: **para que**, **com o objetivo de que**, **conquanto que**.
- 7 Imediatamente antes da palavra “Tolerância” (ℓ.8), pela relação que a oração estabelece com o período anterior, caberia o emprego da conjunção **Entretanto**, seguida de vírgula e letra minúscula.

1 O Governo não é uma entidade abstrata, um
instrumento de coerção ou uma força extrínseca da
4 comunidade nacional. Não é um agente de partidos, grupos,
classes ou interesses. É a própria imagem refletida da pátria
na soma das suas aspirações e no conjunto das suas
7 afinidades. É emanado do povo e, como tal, servo da sua
vontade, provedor de suas necessidades, a força humanizada
e sensível que preside as relações e o desenvolvimento da
sua vida social no sentido da cooperação e da harmonia entre
10 as classes e entre os interesses.

Getúlio Vargas. Discurso de Posse.

Em relação ao texto acima, julgue os itens que se seguem.

- 8 O segmento de texto apresentado é constituído por uma narração.
- 9 O termo “emanado” (ℓ.6) é empregado, no texto, com o sentido de **exercido, realizado, posto em prática**.
- 10 Antes da enumeração iniciada por “servo da sua vontade” (ℓ.6-7), subentende-se a expressão **não pode ser**.

Acerca da elaboração, redação, alteração e consolidação das leis do DF, julgue os seguintes itens.

- 11 A iniciativa comum pode ser exercida pelo governador do DF, por qualquer membro ou órgão da CLDF e pelos cidadãos.
- 12 Procedimento legislativo é o conjunto de atos pré-ordenados que objetivam a formação das leis mediante a colaboração entre os poderes do DF.
- 13 O procedimento legislativo, disciplinado pelo Regimento Interno da CLDF, pode ser ordinário, sumário ou especial.
- 14 No âmbito do Poder Legislativo do DF, denomina-se lei complementar aquela que disciplina matéria que a LODF determina como seu objeto, e resolução, a lei que disciplina, com efeito externo, matéria de competência privativa da CLDF.

No que se refere ao Regimento Interno da CLDF, julgue os próximos itens.

- 15 Os conselhos de representantes comunitários funcionam junto às administrações regionais como órgãos deliberativos, consultivos e fiscalizadores dos atos de gestão dos administradores.
- 16 A denominação Câmara Legislativa decorre da fusão dos nomes atribuídos às casas legislativas dos municípios e dos estados-membros da Federação, respectivamente.
- 17 As sessões legislativas ordinárias têm a duração de nove meses e meio e são divididas em dois períodos, sendo o primeiro iniciado em 1.º de fevereiro e o segundo, em 1.º de agosto.
- 18 Na sessão legislativa extraordinária, a CLDF somente deliberará sobre a aprovação do projeto de lei de diretrizes orçamentárias.
- 19 Cabe à Mesa Diretora organizar a relação dos deputados distritais, que deverá ser concluída após a sessão de posse.
- 20 A criação das regiões administrativas do DF ocorrerá mediante lei aprovada pela maioria simples dos deputados distritais, e sua extinção dar-se-á mediante maioria absoluta.

Quanto à Constituição Federal e aos direitos e garantias fundamentais, julgue os seguintes itens.

- 21 Os direitos e garantias fundamentais integram a proteção constitucional mais importante dirigida aos indivíduos, aos grupos e, em certos casos, até às pessoas jurídicas; apesar disso, esses direitos não têm caráter absoluto, portanto podem sofrer limitações.
- 22 O Brasil adota uma Constituição de tipo rígido e formal, de maneira que as únicas normas de nível constitucional, no país, são a própria Constituição e as emendas constitucionais regularmente aprovadas pelo Congresso Nacional.

Julgue os itens a seguir, acerca da organização do Estado e do Poder Legislativo.

- 23 Uma das mais importantes formas de atuação do Poder Legislativo consiste nas comissões parlamentares de inquérito que se destinam a investigar possíveis atos ilícitos, os quais devem ser determinados para se permitir a instauração válida da comissão. Essas comissões devem funcionar por prazo certo e têm poderes de investigação semelhantes aos das autoridades judiciárias, muito embora determinadas medidas somente possam ser decretadas pelo Poder Judiciário, como é o caso da prisão preventiva.
- 24 Em relação ao DF, é correto afirmar que, embora ele possua as competências reservadas aos estados-membros, não pode ser dividido em municípios; ele é regido por lei orgânica votada em dois turnos por sua Câmara Legislativa, com intervalo mínimo de dez dias entre as votações.

Em relação à tributação e às finanças públicas, julgue os itens seguintes.

- 25 É privativa da União a competência para emitir moeda, a qual é exercida pela Casa da Moeda do Brasil.
- 26 No que tange à repartição das competências tributárias, a competência para instituir taxa será da pessoa jurídica de direito público que preste serviço de qualquer natureza ao qual a taxa esteja associada, ou que exerça poder de polícia do qual a taxa decorra.
- 27 Em virtude do princípio constitucional da legalidade, apenas a lei em sentido formal, isto é, aquela aprovada pelo Poder Legislativo após o devido processo, pode criar ou majorar tributos; em consequência, o Poder Executivo não pode fazê-lo por meio de medida provisória.

Julgue os itens que se seguem, a respeito da LODF e da organização do DF.

- 28 Um indivíduo não-nascido no Brasil não pode ser eleito governador do DF, à luz das disposições da LODF.
- 29 De acordo com a LODF, um dos objetivos prioritários do DF consiste em preservar o conjunto urbanístico de Brasília, de suas cidades-satélites e do chamado Entorno do DF, de maneira a se preservar sua identidade, adequando-se as exigências do desenvolvimento à defesa de sua memória, de sua tradição e de suas peculiaridades.
- 30 Se, em determinado ano, um anteprojeto de lei for rejeitado pela CLDF, a matéria dele constante poderá ser reapresentada a esse órgão na mesma sessão legislativa, desde que a maioria absoluta dos membros da casa assim proponha; por outro lado, se o veto do Poder Executivo a um projeto de lei for derrubado pela CLDF, o projeto poderá vir a ser promulgado pelo próprio presidente da Câmara, caso o governador não o faça dentro de certo prazo.

No atinente aos atos administrativos, ao controle da administração pública, aos contratos administrativos e aos agentes administrativos, julgue os itens subseqüentes.

- 31 A atuação de particulares em colaboração com o poder público pode ocorrer até independentemente de remuneração, bem como em situações nas quais a remuneração não é paga pelo ente público, mas por pessoas jurídicas de direito privado.
- 32 Para, pelo menos, parte da doutrina especializada, não obstante o dever de legalidade da administração pública, esta pode, em casos excepcionais e observadas certas condições, deixar de declarar a invalidade de um ato administrativo, quando constatar e justificar cabalmente que a anulação causará mais mal que a manutenção do ato.
- 33 Uma das modalidades mais relevantes de controle da administração pública é o realizado por ela própria, o qual decorre do poder de autotutela dos entes e órgãos públicos; esse controle interno fundamenta-se no princípio da legalidade e pode ser tanto realizado de ofício quanto mediante provocação do interessado; pode, ainda, ser preventivo, concomitante ou posterior.
- 34 Devido à presunção de validade dos atos da administração pública e à formalidade a eles inerente, os contratos administrativos concernentes a direitos reais sobre imóveis não precisam ser lavrados em serviço notarial.

Julgue os itens a seguir, no que tange aos poderes e princípios básicos da administração e aos serviços públicos.

- 35 De acordo com Celso Antônio Bandeira de Mello, o princípio da adaptabilidade implica a atualização e a modernização dos serviços públicos, ainda que isso ocorra apenas em função das possibilidades materiais do poder público.
- 36 O poder regulamentar é a única maneira pela qual se exerce a função normativa do Poder Executivo.
- 37 Por força do princípio do devido processo legal, a administração deve sempre comunicar ao interessado qualquer decisão que lhe afete a propriedade ou a liberdade; essa comunicação deve ser feita, em regra, previamente à prática do ato, mas poderá ser posterior, em situações excepcionais.

Referentemente à Lei de Licitações e Contratos Administrativos (Lei n.º 8.666/1993), à Lei n.º 8.112/1990, da União, na forma consolidada pelo Decreto Legislativo n.º 1.094/2004, e à Lei de Permissões e Concessões, julgue os itens seguintes.

- 38 Uma das características essenciais das concessões e permissões de serviços públicos consiste em que a remuneração do prestador do serviço se faz, necessariamente, mediante a cobrança de tarifa dos usuários, de maneira a não onerar o Estado pela prestação da atividade.
- 39 As sociedades de economia mista e empresas públicas exploradoras de atividade econômica não estão obrigadas a licitar.
- 40 Nos termos do Decreto Legislativo n.º 1.094/2004, que consolidou a aplicação, no DF, da Lei n.º 8.112/1990, da União, a responsabilidade do servidor do DF pelo cometimento de ilícito pode ser, de modo cumulativo, de natureza cível, penal e administrativa, e de modo relativamente independente; no caso de o agente público cometer ato sujeito à pena de demissão, esta pode ser aplicada ainda que o servidor não tenha sofrido sanção anterior.

Read the following text to answer items 41 through 50.

Software engineering fundamentals

1 As long as software quality (reliability and
availability attributes in particular) remains suspect, there
will be a reluctance to use software in safety-critical
4 applications involving a high cost of failure. However, there
is also strong pressure from other quarters to use software
in critical applications because of cost and technical
7 advantages; this pressure drives software developers to find
ways to produce software with a high reliability.

Safety considerations are also the domain of quality
10 assurance. Some of the tools employed in software safety
analyses include failure modes and effects analysis (FMEA),
fault tree analysis, event tree analysis, hazard analysis, and
13 Petri nets. Usually these analyses include comprehensive
models of hardware, software, and operator performance.

The objective of FMEA is to analyze an information
16 system's components systematically to determine how a
component might fail and evaluate the effect of such a failure
on system performance under a variety of operational
19 conditions. The objective is to identify ways in which
combinations of external events, operator reactions, and/or
hardware/software failures can lead to serious failure
22 consequences. Some analytical approaches (fault tree
analysis or event tree analysis, for example) begin by
assuming the occurrence of a catastrophic failure and
25 working backwards to find out what logical sequence of
events would result in such a failure.

A. Behforooz and F. J. Hudson. *Software engineering fundamentals*.
Oxford University Press, Inc., 1996, p. 554 (with adaptations).

Based on the text above, judge the following items.

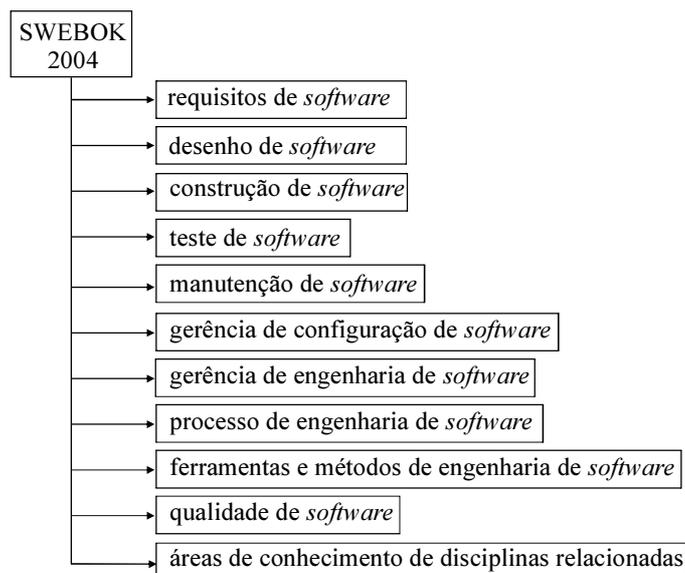
- 41 Analytical software safety approaches take into account the fault tree analysis or the event tree analysis only.
- 42 Software quality depends basically on two main factors.
- 43 Reliability is one of the software developers main concerns.
- 44 As long as software quality is under suspicion everybody refuses to use it.
- 45 Safety considerations have nothing to do with quality assurance.
- 46 Serious failure consequences can be a combination of external factors and hardware/software failures.

In the text above

- 47 "to find out" (l.25) means **to obtain knowledge of something**.
- 48 "to use" (l.3) is the same as **to wear**.
- 49 "high" (l.8) is synonymous with **tall**.
- 50 "comprehensive" (l.13) means **covering completely or broadly**.

CONHECIMENTOS ESPECÍFICOS

Texto para os itens de 51 a 61.



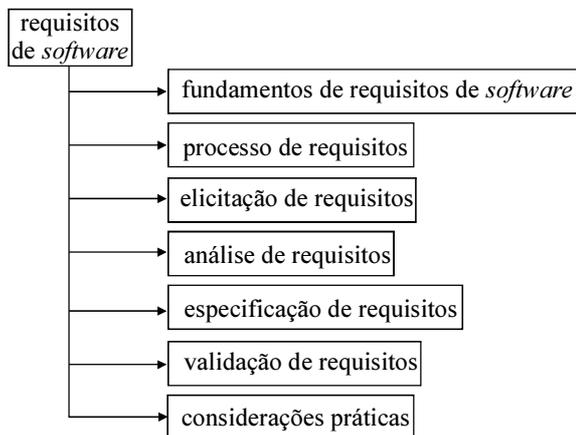
Os especialistas vêm discutindo, atualmente, os métodos ágeis de desenvolvimento de *software*, com especial ênfase em programação extrema (XP), que é caracterizada por uma série de práticas que promovem a produção de *software* de alta qualidade, ao mesmo tempo em que maximiza a adaptabilidade da equipe de desenvolvimento a mudanças nos requisitos. Dizem os especialistas que o XP é capaz de atingir um elevado nível de dinamismo no desenvolvimento e de qualidade no *software* desenvolvido por meio de práticas como: interação freqüente com o cliente; testes automatizados; desenvolvimento incremental e integração contínua; programação pareada; refatoração; padronização de codificação; iterações curtas etc. Diversas são as experiências de sucesso no uso de XP, como a da Assembléia Legislativa, onde está sendo desenvolvido um novo sistema para gerenciamento de recursos humanos.

A partir das informações do texto acima e considerando os conceitos do SWEBOK, cuja estrutura está resumida na figura contida no texto, julgue os itens a seguir.

- 51** Segundo o texto, uma das grandes vantagens da programação extrema é a redução de mudanças nos requisitos de *software* e sistemas por meio de práticas como interação freqüente com o cliente e iterações curtas.
- 52** O SWEBOK é um conjunto de definições de normas e padrões, criado para guiar o desenvolvimento de *software* com qualidade.
- 53** No desenvolvimento de *software*, mudanças em requisitos são comuns e muitas vezes inevitáveis, especialmente quando se trata em desenvolver *software* para o suporte a processos de trabalho em organizações.

Ainda considerando as informações do texto, julgue os itens seguintes.

- 54** O emprego de verificação & validação (V&V) integrada confere elevado grau de qualidade ao desenvolvimento de *software*. As práticas do XP de interação freqüente com o cliente, de desenvolvimento incremental, integração contínua, programação pareada e iterações curtas são práticas formais de V&V.
- 55** Se a prática de testes automatizados for embasada na construção e na execução automática de módulos de teste unitário, a adoção de linguagens orientadas a objetos favorece a redução nos custos da prática de testes.
- 56** Se a prática de integração contínua for embasada no uso de ferramentas que automatizam a construção de *builds* para posterior execução de testes automatizados, depreende-se que a prática de integração contínua é diretamente relacionada à disciplina de gerência de configuração de *software*, no que se refere ao gerenciamento de liberação e entrega de *software*.
- 57** A prática de refatoração está mais fortemente relacionada à manutenção de *software*, e menos relacionada ao *design* de *software*.
- 58** As práticas de desenvolvimento incremental e de iterações curtas estão mais associadas ao modelo de ciclo de vida em espiral que ao modelo de ciclo de vida em cascata, inclusive devido à maior aplicabilidade à gerência de riscos daquele que deste.
- 59** A padronização de codificação é mais fortemente relacionada à construção de *software* que ao desenho de *software*.
- 60** De acordo com o texto, o uso de XP estabeleceu a melhoria contínua do processo de produção de *software* na Assembléia Legislativa.
- 61** As informações contidas no texto são suficientes para se concluir que o novo sistema para gerenciamento de recursos humanos da Assembléia Legislativa consiste em um novo sistema de informações.



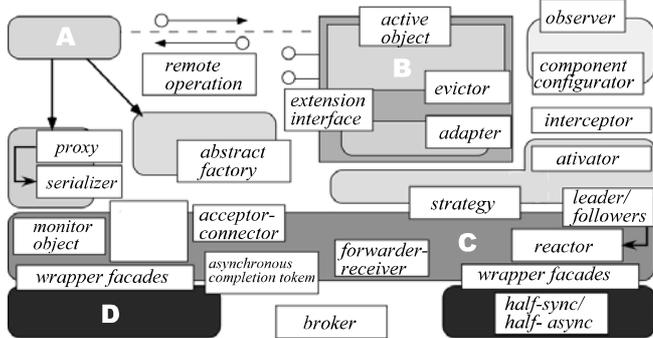
A figura acima apresenta uma lista de tópicos abordados na área de conhecimento de requisitos de *software* do SWEBOK. Julgue os itens seguintes, relativos às informações apresentadas, aos conceitos do SWEBOK, à análise de negócios e requisitos de *software* e à especificação de requisitos de *software*.

- 62** Considerando-se que a prática de interação freqüente com o cliente existente no método XP provoca *feedback* imediato para o cliente acerca do impacto de possíveis mudanças nos requisitos sobre os custos e prazos de um projeto, é correto considerar que tal prática permite estabelecer um modelo para a negociação imediata de requisitos.
- 63** Segundo o SWEBOK, requisitos de *software* e requisitos de sistema são sinônimos, bem como desenvolvimento de *software* e desenvolvimento de sistemas são também sinônimos.
- 64** A análise de negócios pode ser considerada uma atividade da fase de análise de requisitos e, desse modo, gera demandas para a fase anterior, de elicitação de requisitos.
- 65** O processo de requisitos de *software*, composto pelas fases de elicitação de requisitos, análise de requisitos, especificação de requisitos, validação de requisitos e gestão de mudanças de requisitos, é executado apenas durante a fase inicial de um projeto de desenvolvimento de *software*.
- 66** No âmbito de um projeto de desenvolvimento de *software*, a construção de um modelo conceitual formalizado, com definição genérica de termos e relações entre termos da área de gestão de recursos humanos, é uma prática mais relacionada com a análise de requisitos e menos relacionada com a especificação de requisitos.
- 67** O contato constante com o cliente, durante o desenvolvimento de *software*, a fim de identificar junto ao mesmo se as funcionalidades estão sendo adequadamente implementadas, é uma atividade de validação de requisitos.

		propósito		
		criação	estrutura	comportamento
escopo	classe	<i>factory method</i>	<i>class adapter</i>	<i>interpreter</i> <i>template method</i>
	objeto	<i>abstract factory</i> <i>builder</i> <i>prototype</i> <i>singleton</i>	<i>object adapter</i> <i>bridge</i> <i>composite</i> <i>decorator</i> <i>facade</i> <i>flyweight</i> <i>proxy</i>	<i>chain of responsibility</i> <i>command</i> <i>iterator</i> <i>mediator</i> <i>memento</i> <i>observer</i> <i>state</i> <i>strategy</i> <i>visitor</i>

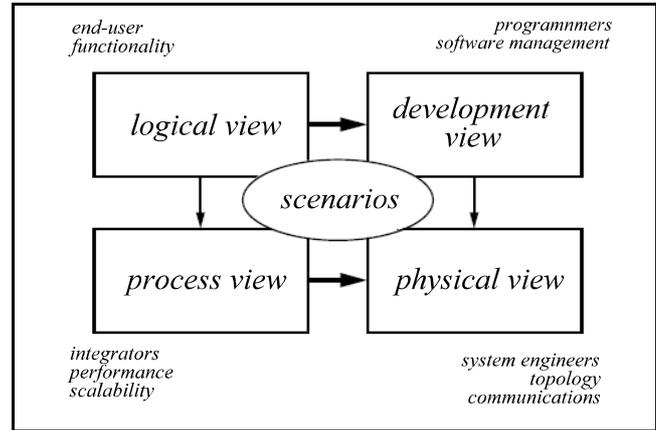
A figura acima, extraída do sítio <http://www.argonavis.com.br>, apresenta uma classificação de um conjunto de padrões de desenho orientado a objetos, também denominados padrões de projeto. Os padrões apresentados são conhecidos como GOF *patterns* (*gang-of-four patterns*). A respeito das informações apresentadas e dos conceitos de padrões de desenho orientado a objetos, julgue os itens a seguir.

- 68** A implementação de uma lista encadeada de filtros para tratamento de pedidos e respostas do protocolo http pode empregar o padrão *chain of responsibility*.
- 69** Com relação aos padrões apresentados na figura, cada padrão de desenho é mutuamente complementar a todos os outros padrões, não havendo conflitos na indicação de uso e aplicabilidade entre eles.
- 70** Os padrões de desenho apresentados podem ser enquadrados em criacionais, comportamentais e estruturais, sendo os padrões comportamentais orientados à dinâmica de criação de estruturas de dados orientadas a objeto, enquanto que os estruturais são orientados para o desenho de algoritmos orientados a objetos.
- 71** Quando se deseja encapsular e tornar intercambiáveis um conjunto de implementações alternativas de algoritmos que utilizam os mesmos conjuntos de parâmetros, sugere-se adotar o padrão *strategy*, em vez do padrão *chain of responsibility*.
- 72** O padrão *abstract factory* envolve o emprego coletivo do padrão *factory method*.
- 73** O padrão *facade* pode ser combinado com o uso do padrão *singleton*, já que, em geral, as fachadas são instâncias únicas de sistemas orientados a objeto.
- 74** Entre os padrões apresentados, o mais comumente usado durante a construção de *frameworks* é o *template method*.
- 75** A construção de *menus* com o usuário, em aplicativos interativos e orientados a objeto, faz uso explícito do padrão *iterator*.
- 76** Quando um analisador sintático de expressões matemáticas é construído, usando-se padrões de desenho, indica-se o emprego dos padrões *iterator*, *memento*, *observer* e *state*, em vez dos padrões *visitor*, *interpreter* e *composite*.
- 77** Na definição da estrutura de diretórios e arquivos em um sistema de arquivos convencional de um computador, é correto o uso do padrão *composite*.
- 78** O tratamento de coleções uniformes de dados pode ser aprimorado diretamente por meio do uso do padrão *mediator* e, em menor escala, por meio do uso do padrão *iterator*.
- 79** A microarquitetura de programas que realizam tratamento de eventos na interface com o usuário usualmente emprega o padrão *observer*.
- 80** O emprego do padrão *decorator* permite a atribuição dinâmica de responsabilidades a um objeto já instanciado, além de permitir preservar as responsabilidades do objeto já instanciado.
- 81** O suporte ao desfazer (*undo*) de operações em editores interativos pode ser usualmente efetuado por meio do emprego do padrão *memento*.



A figura acima, que foi adaptada do sítio <http://www.cs.wustl.edu>, apresenta um sistema de padrões de desenho conhecido como POSA2, constituído, entre outros, pelos elementos indicados por A, B, C e D e pelos padrões de desenho orientado a objeto nomeados nos retângulos mostrados. Com relação às informações apresentadas e aos conceitos de análise e projeto de *software* orientado a objetos, de arquitetura de *software*, de programação orientada a objetos e de padrões de projeto ou desenho, julgue os itens que se seguem.

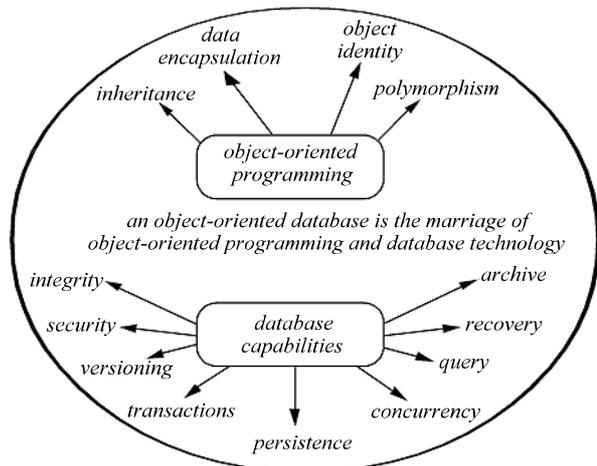
- 82 Para a alocação e a de-alocação de componentes de *software* sob demanda, quando da invocação de serviços, o padrão *activator* é mais adequado que o padrão *extension interface*.
- 83 Todos os padrões de desenho orientado a objetos mostrados estão no nível microarquitetural.
- 84 Sabendo-se que foi estabelecida uma relação cliente-servidor entre os elementos A e B, é correto afirmar que A é o cliente e B, o servidor.
- 85 Sabendo-se que um dos elementos A, B, C ou D desempenha o papel de *middleware*, é correto afirmar que o elemento C é o elemento que desempenha esse papel.
- 86 Caso seja necessário adotar uma solução reusável para controlar o acesso concorrente a um componente de *software*, é mais indicado o uso do padrão *broker*, em comparação com o padrão *monitor object*.
- 87 Caso seja necessário prover acesso orientado a objetos a um conjunto de operações disponíveis em um sistema legado, entre os padrões *adapter* e *wrapper facades*, o uso do primeiro é mais adequado.
- 88 Caso se deseje transmitir, por meio de invocação de procedimento remoto orientado a objeto, uma lista de parâmetros. O padrão *acceptor-connector* é mais indicado que o padrão *serializer*.



Kruchten. Architectural blueprints – the 4+1 view model of *software* architecture.

A figura acima apresenta o modelo denominado visão 4+1 utilizado para a descrição da arquitetura de sistemas de *software*. Na periferia dos quatro quadrantes do diagrama ilustrado, são apresentados exemplos de tópicos de discussão, conceitos e profissionais associados à construção de cada visão. No que se refere aos conceitos e informações apresentadas na figura acima e à sua relação com os elementos da estrutura e dos requisitos de *software* do SWEBOK, dos padrões GOF e do sistema POSA2, julgue os itens subsequentes.

- 89 As setas apresentadas no diagrama estabelecem uma ordenação temporal na construção das visões arquiteturais e, desse modo, o modelo de visão arquitetural 4 + 1, de uma forma geral, possui uma forte associação com as disciplinas de processo de engenharia de *software* e gerência de engenharia de *software* do SWEBOK, o que não ocorre com os sistemas de arquitetura estabelecidos no GOF e no POSA2 que por sua vez estão mais orientados para a disciplina de desenho de *software* do SWEBOK.
- 90 A visão lógica (*logical view*) está mais associada à disciplina de requisitos de *software* que a de construção de *software* do documento SWEBOK.
- 91 Com relação ao SWEBOK, a visão de desenvolvimento (*development view*) está mais associada à disciplina de gerência de engenharia de *software* que à de qualidade de *software*.
- 92 Entre as disciplinas de processo de engenharia de *software* e de desenho de *software* do SWEBOK, a primeira está mais associada à visão de processo (*process view*).
- 93 A visão física (*physical view*) está mais associada à disciplina de qualidade de *software* que à de gerência de configuração de *software*, ambas do documento SWEBOK.



Internet: <<http://people.morehead-st.edu>>.

A figura acima relaciona conceitos oriundos dos paradigmas de base de dados e de programação orientada a objetos. Julgue os próximos itens, considerando os elementos apresentados nessa figura.

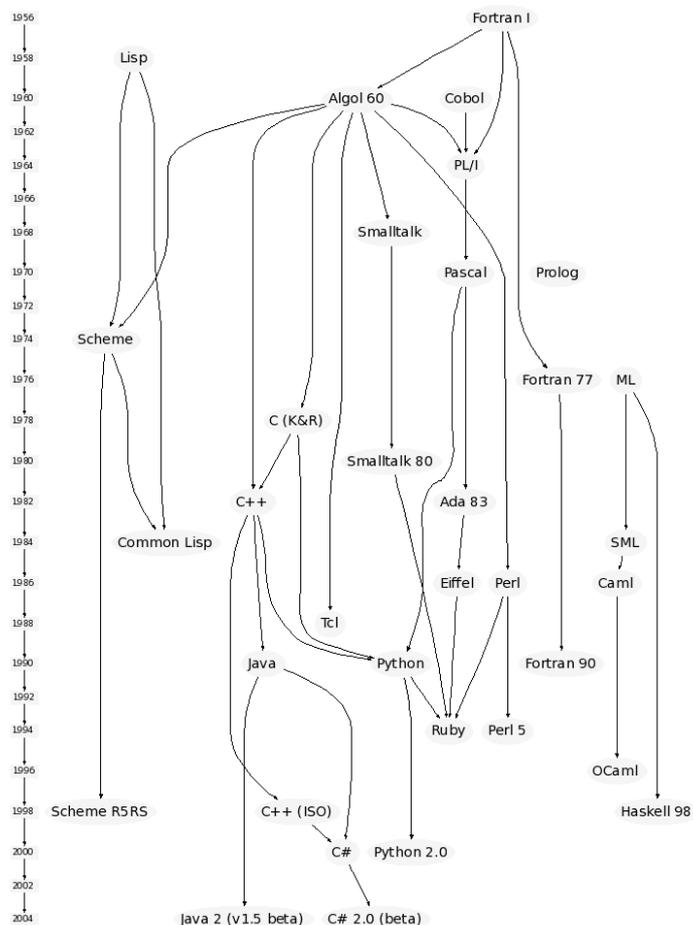
94 As demandas de tratamento de integridade, concorrência e recuperação durante a implementação de SGBDs são maiores nos casos de banco de dados que fornecem suporte a transações, quando comparadas com aqueles que trabalham com bases de dados analíticas e com tecnologias de *datawarehouse*.

95 SGBDs de código aberto, como o MySQL, suportam o armazenamento persistente de cadeias de *bytes* que foram geradas a partir de objetos Java que estavam alocados na memória *heap*, e que foram serializados usando-se algoritmos nativos da JVM. Para o armazenamento de objetos serializados em SGBD de código aberto, é indicado o uso de tipos de dados SQL BLOB ou outro tipo de dados de tamanho variável que armazene cadeias de caracteres binários.

96 O encapsulamento de dados facilita a implementação do mapeamento objeto-relacional.

97 O emprego de herança, especialmente quando usada em classes de objetos persistentes, facilita o mapeamento objeto-relacional.

Texto para os itens de 98 a 106.



Internet: <<http://www.cs.umd.edu>>.

O diagrama acima apresenta alguns relacionamentos estabelecidos entre linguagens de programação concebidas nos últimos cinquenta anos.

Com referência às informações apresentadas na figura contida no texto e aos conceitos de linguagens de programação, especialmente presentes nas linguagens Java, Delphi, PHP, C, C++, Perl, Python, Lisp, AWK e C#, julgue os itens a seguir.

98 PHP é uma linguagem de *scripting* que descende diretamente da linguagem PERL. Do mesmo modo que Python, PERL e PHP suportam operações sobre *arrays* associativos, o que facilita operações básicas de tratamento de estruturas de dados indexadas. PHP, adicionalmente, oferece maior suporte a operações sobre o protocolo http e à linguagem HTML, quando comparada com a linguagem PERL padrão.

99 A linguagem de programação Java, cuja sintaxe deriva diretamente de C++, apresenta um forte sistema de tipos de dados, inclusive com verificação de tipos em *runtime*, diferentemente do que ocorre com suas antecessoras, as linguagens C e C++.

100 A linguagem Python é do tipo orientada a objetos e fornece suporte a herança, polimorfismo e ligação dinâmica. A linguagem também se caracteriza por possuir capacidade de integração com bibliotecas nativas escritas em C.

Ainda com relação às informações apresentadas no texto e aos conceitos de linguagem de programação, julgue os itens seguintes.

- 101** Comandos básicos de controle de fluxo, como `repeat`, `loop` e `for`, estabelecidos desde a linguagem C, estão também presentes nos dialetos comuns de LISP.
- 102** A linguagem ou tecnologia comumente conhecida pelo nome Delphi está diretamente associada com o desenvolvimento de uma versão orientada a objetos da linguagem Pascal, mas esse aperfeiçoamento tornou a linguagem Delphi uma linguagem com sistema de tipos de dados que é fraco.
- 103** As linguagens C, C++ e Java podem ser consideradas descendentes ou ascendentes diretas, na seqüência apresentada, visto que todos os comandos e funções da linguagem C padrão estão disponíveis na linguagem C++ e todos os comandos e funções da linguagem C++ estão disponíveis na linguagem Java.
- 104** A programação na linguagem LISP é embasada em tratamento de listas, árvores, recursividade e funções de alta ordem, e cujos fundamentos teóricos estão no cálculo Lambda.
- 105** Programas na linguagem AWK tendem a ter um tamanho de várias dezenas de linhas de código, mas, devido à disponibilidade de mecanismos de modularidade, especialmente derivados da linguagem C, mantêm a propriedade de serem facilmente mantidos.
- 106** A linguagem C#, cuja sintaxe é fortemente embasada na linguagem Java, possui um amplo conjunto de bibliotecas para manipulação de tipos de dados complexos, mas não dispõe de mecanismos para tratamento de grandes módulos de *software* nem gerenciamento automático de memória, como ocorre com a linguagem Java.

```
1 public class Example {
2     public static int horner (int [] a, int n, int r) {
3         int result = a [n];
4         for (int i = n-1; i >= 0; --i)
5             result = result * r + a[i];
6         return result;
7     }
8     public static void main(String[] args) {
9         int[] a = {1, 2, 3, 4};
10        System.out.println (horner(a, 3, 3));
11    }
12 }
```

Internet: <<http://www.brpreiss.com>>.

O código acima apresenta a implementação de um algoritmo chamado *horner*, na linguagem Java. A partir desse algoritmo e considerando que o método `main` da classe `Example` seja executado, julgue os itens que se seguem.

- 107** A ordem de complexidade temporal do algoritmo *horner* é linear.
- 108** Após a referida execução, o número 142 será impresso.

Segundo o PMBoK, qualidade é a totalidade de características de uma entidade relacionadas à habilidade de satisfazer necessidades expressas ou implicadas. A qualidade é um objetivo que se busca tanto para garantir níveis adequados de satisfação como forma de diferencial competitivo. Acerca dos processos de gestão de qualidade, julgue os itens que se seguem.

- 109** Para se assegurar a qualidade, as atividades planejadas e sistemáticas devem ser implementadas em um sistema de qualidade que reflita os padrões de qualidade que se deseja atingir. Esse processo tem por objetivo prover confiança de que um projeto deve atingir os padrões de qualidade estabelecidos.
- 110** Um aspecto crítico para a gestão de qualidade consiste em tornar necessidades implícitas em necessidades expressas por meio da gestão de escopo.

Os sistemas de *software* tiveram tradicionalmente com o foco a funcionalidade, tendo sido dada pouca importância à segurança da informação em seu desenvolvimento. O desenvolvimento de *software* seguro vem aparecendo nos últimos anos como uma disciplina importante que altera essa perspectiva clássica. Acerca da segurança em sistemas de *software*, julgue os itens a seguir.

- 111** Uma das principais preocupações de segurança em *software* aplicativo é a utilização de senhas pessoais como forma de autenticação de usuários. Como uma regra geral, as senhas pessoais não podem ser armazenadas em claro em bancos de dados ou mesmo em instâncias de *software* mantidas em memória durante a execução do *software*. Uma maneira simples de solucionar esse problema consiste na utilização de funções *hash* unidirecionais, que são aplicadas às senhas tão logo elas sejam digitadas, gerando uma mensagem de resumo que é armazenada em banco de dados. Entretanto, esse sistema ainda é suscetível a ataques de dicionário, permitindo a quebra de senhas frágeis.
- 112** Defeitos em *software* que acarretam o estouro de *buffer* podem permitir que atacantes experientes executem códigos maliciosos em sistemas que executam *software* que contenha o referido defeito.
- 113** A interceptação de mensagens trocadas em claro através da rede entre entidades de uma aplicação distribuída consiste em um ataque capaz de comprometer a confidencialidade dos dados envolvidos nas transações, mesmo que a comunicação entre as entidades se dê por intermédio de um protocolo de comunicação proprietário e desconhecido pelo atacante.

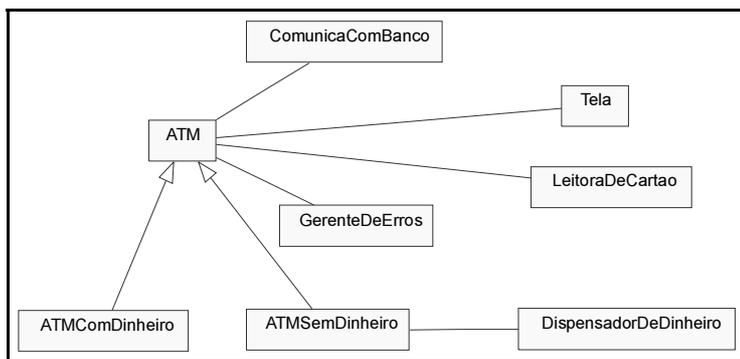


Figura I

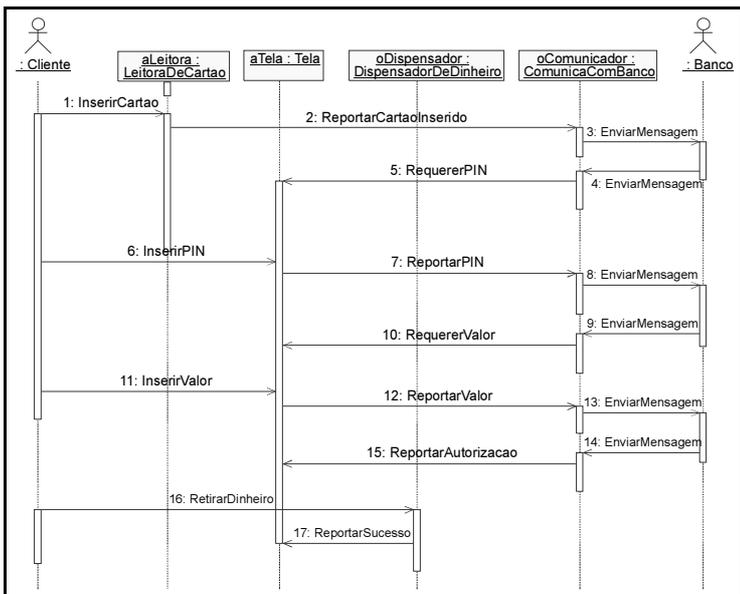


Figura II

A figura acima apresenta um conjunto de diagramas UML simplificados de um sistema de *software* para terminais de auto-atendimento bancário (ATM). Com base nesses diagramas, julgue os próximos itens.

114 Os objetos que possuem nomes na figura II são instâncias únicas de suas respectivas classes nesse sistema.

115 O diagrama de seqüências da figura II pode ser convertido em um diagrama de colaboração, sem perda da informação contida em sua representação.

116 Na figura I, as informações relativas à multiplicidade nos relacionamentos de associação foram omitidas.

O processo unificado (RUP) tem por objetivo estabelecer um *framework* para desenvolvimento iterativo de *software*. Acerca do RUP, julgue os itens a seguir.

117 Cada uma das fases de desenvolvimento do RUP envolve certa quantidade de esforço em atividades relacionadas com as diversas disciplinas de desenvolvimento de *software*. O esforço em gerência de configuração e de mudanças é maior nas fases de inspeção e elaboração, diminuindo à medida que o processo se aproxima das fases de construção e transição. Por outro lado, o esforço na realização de testes de *software* é mínimo durante as fases iniciais e aumenta consideravelmente na fase de construção.

118 A definição das tarefas executadas em cada iteração está diretamente relacionada à priorização de casos de uso. Assim, os casos de uso mais simples são executados primeiro, enquanto os casos de uso de maior complexidade ficam para ser executados nas iterações mais avançadas, quando o conhecimento sobre o problema a ser resolvido é maior.

Existem, hoje, diversas ferramentas de *software* livre disponíveis para apoiar o desenvolvimento de aplicações de *software*, assim como para compor o seu ambiente de operação. Acerca dos ambientes de desenvolvimento e de operação de *software* que utiliza ferramentas de *software* livre, julgue os itens a seguir.

119 Uma ferramenta muito utilizada para o controle de versões em ambientes de desenvolvimento em time é o pacote CVS (*concurrent version system*), que possui diversas implementações de cliente disponíveis sob licenças consideradas de *software* livre, tais como as licenças GNU e BSD.

120 Um exemplo de ambiente de produção para aplicações com arquitetura em três camadas totalmente composto de *software* livre é: camada de banco de dados implementada com um SGBDR (sistema gerenciador de banco de dados relacional) PostgreSQL; camada de negócios implementada com o servidor de aplicações TomCat e camada de cliente implementada com um navegador Firefox que executa aplicativos Java.

PROVA DISCURSIVA

- Nesta prova — que vale **dez** pontos —, faça o que se pede, usando o espaço indicado no presente caderno para rascunho. Em seguida, transcreva o texto para a **FOLHA DE TEXTO DEFINITIVO DA PROVA DISCURSIVA**, no local apropriado, pois **não serão avaliados fragmentos de texto escritos em locais indevidos**.
- Qualquer fragmento de texto que ultrapassar a extensão máxima de **trinta** linhas será desconsiderado.
- **ATENÇÃO!** Na **folha de texto definitivo**, identifique-se apenas no cabeçalho da primeira página, pois **não será avaliado** texto que tenha qualquer assinatura ou marca identificadora fora do local apropriado.

Diariamente, passam pela Câmara cerca de mil pessoas. Telefonemas e correspondências para a presidência da Casa e para os gabinetes dos deputados ampliam o contato com o público que busca, no jovem Legislativo local, respostas para seus anseios.

A Câmara foi estruturada de forma a garantir suporte à atuação legislativa (elaboração e discussão de leis) dos deputados. Além disso, a Casa precisa assegurar, com eficiência e rapidez, a administração interna e o atendimento à população que transita diariamente por suas dependências, em uma rotina comum às casas legislativas.

O coração político do legislativo distrital é o plenário, palco das discussões e votações.

A administração fica a cargo da presidência, da vice-presidência e de três secretarias, comissões e assessorias de apoio ao processo legislativo.

O prédio da Câmara Legislativa abriga os setores administrativos, as assessorias técnicas, o serviço médico, o Fundo de Assistência à Saúde (Fascal), postos do Banco de Brasília, Banco do Brasil e agência dos Correios.

Essa estrutura é mantida por cerca de mil e quinhentos funcionários, que se dividem em servidores efetivos (concursados) e servidores de livre provimento, nomeados pelos deputados. A casa abriga as seguintes comissões permanentes: CCJ – Comissão de Constituição e Justiça; COF – Comissão de Orçamento e Finanças; CDDHCEDP – Comissão de Defesa dos Direitos Humanos, Cidadania, Ética e Decoro Parlamentar; CAS – Comissão de Assuntos Sociais; CDC – Comissão de Defesa do Consumidor; CAF – Comissão de Assuntos Fundiários; CES – Comissão de Educação e Saúde; CS – Comissão de Segurança; e CDESCTMAT – Comissão de Desenvolvimento Econômico Sustentável, Ciência, Tecnologia, Meio Ambiente e Turismo.

Internet.: <<http://www.cl.df.gov.br>> (com adaptações).

Tendo por base a descrição da Câmara Legislativa do DF apresentada acima, esboce um diagnóstico fictício da condição atual dos sistemas de tecnologia da informação e *software* da referida casa e proponha um programa de aperfeiçoamento em engenharia de *software* e sistemas para a CLDF, considerando que a descrição apresentada faça parte de um estudo aprofundado sobre a organização. O diagnóstico e o programa de aperfeiçoamento devem, necessariamente, contemplar os seguintes aspectos:

- ▶ métodos e processos de engenharia de *software*;
- ▶ engenharia de requisitos de *software* e sistemas de informação;
- ▶ processos organizacionais legislativos e administrativos;
- ▶ arquitetura de *software* e padrões de projeto (*design*);
- ▶ gerenciamento de *software*, informação e bases de dados usando tecnologia de *software* livre, de código aberto e linguagens orientadas a objeto.

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