1. Sobre temporizadores indique a alternativa INCORRETA
   1. São necessários para a implementação de tarefas periódicas
   2. É capaz de gerar interrupções com uma frequência configurável
   3. Utilizando um temporizador é possível calcular qualquer intervalo de tempo
   4. Estão presentes na maioria da aplicações de tempo real
   5. A resolução de um temporizador é o tempo entre as interrupções geradas pelo mesmo

Resposta: C

1. Sobre as diferenças entre um sistema operacional de propósito geral (SOPG) e um sistema operacional de tempo real (SOTR) é CORRETO afirmar que
   1. O SOPG não costuma utilizar mecanismos que melhorem o desempenho médio do sistema
   2. Um exemplo de mecanismo interessante para se utilizar num SOTR é a memória virtual
   3. Um SOPG tem como principal objetivo atender todas as deadlines das tarefas.
   4. A utilização de memória cache não é interessante para um SOTR
   5. Um SOTR busca uma distribuição uniforme dos recursos

Resposta: D

1. Marque (V) para as alternativas corretas e (F) para as incorretas

( ) É interessante fazer um teste de escalonabilidade para aplicações de tempo real

( ) Um sistema operacional de tempo real (SOTR) busca sempre melhorar o desempenho médio do sistema

( ) Os serviços providos por um SOTR são definidos apenas pelos seus aspectos funcionais

( ) É interessante que um SOTR dê suporte a concorrência

( ) Não existe vantagens em se utilizar de um sistema operacional de propósito geral para aplicações de tempo real

Resposta : VFFVF

1. Qual das características a seguir são interessantes para uma aplicação de tempo real
   * 1. Suporte a concorrência
     2. Uso de memória cache
     3. Previsibilidade temporal
     4. Uso de memória virtual
2. i,ii e ii
3. i e iv
4. i e iii
5. ii,iii e iv
6. ii e iii

Resposta: C

1. Sobre tarefas e threads, assinale a alternativa INCORRETA
   1. O uso de threads visa diminuir o tempo de mudança de contexto
   2. O estado “PRONTO” representa que a thread está pronta para ser destruída
   3. Uma thread herda diversos atributos de uma tarefa hospedeira
   4. O Contexto de execução é um exemplo de atributo que uma tarefa possui
   5. Variavéis compatilhadas é uma técnica para comunicação entre tarefas e threads

Resposta: B

1. Qual a melhor abordagem para resolver o problema de tratamento de interrupções quando o kernel está executando?
   1. O kernel desabilita as interrupções e o tratador só é ativado quando a tarefa em execução deixa o kernel
   2. O kernel tem pontos de preempção previamente programados
   3. O kernel é completamente preemptivo menos quando houver modificações em estruturas de dados compartilhadas
   4. O kernel é completamente preemptivo a qualquer momento, sem preocupações com a consistencia das estruturas compartilhadas
   5. Nenhuma das alternativas

Resp: C

1. Marque a opção que NÃO consiste em uma métrica de SOTR:
   1. Latência de Chaveamento de Contexto
   2. Latência de Interrupção
   3. Tempo de execução máximo de chamadas ao sistema
   4. Maior intervalo de tempo que as interrupções podem ficar desabilitadas
   5. Latência de paginação da memória virtual

Resp: E

1. Marque a opção que NÃO consiste em um requisito de SOTR:
   1. Dar suporte a multi-thread com escalonamento baseado em prioridades e preempção
   2. Dar suporte a memória virtual
   3. Incluir um mecanismo de sincronização de threads com comportamento previsível
   4. Deve haver um mecanismo para previnir a inversão de prioridades
   5. O comportamento do SO em termos de métricas deve ser conhecido e previsível, para todos os cenários possíveis

Resp: B

1. Dentre as seguintes características, qual a mais importante para um SOTR?
   1. Compatibilidade
   2. Facilidade de programação
   3. Desempenho
   4. Previsibilidade
   5. Nenhuma das alternativas

Resp: D

1. Assinale Verdadeiro (V) ou Falso (F) sobre SOTR:

( ) Em um SOTR as interrupções nunca podem estar desabilitadas

( ) Um requisito para SOTR é que deve ser possível preemptar o kernel a qualquer momento

( ) Latência de interrupção equivale ao release jitter da rotina tratadora

( ) Tratamento de inversão de prioridades é um fator importante mas não necessário num SOTR

( ) Técnicas como garbage collection não são implementadas em aplicações de tempo real por ter um comportamento imprevisível

Resp: FFVFV

1. (Tipos de Suporte de Tempo Real) Assinale a alternativa **verdadeira**:
   1. Núcleos de Tempo Real (NTR) possuem como características principais uma alta previsibilidade temporal e um conjunto robusto de serviços oferecidos pelo kernel.
   2. O escalonamento baseado em prioridades é comumente adotado tanto em NTRs quanto em Sistemas Operacionais de Tempo Real (SOTR).
   3. O kernel de um NTR oferece um nível maior de abstração das chamadas do sistema ao programador que um SOTR.
   4. Um micro-kernel apenas é capaz de suprir os requisitos de um NTR, mas não de um SOTR.
   5. Um produto divulgado como um SOTR implica na existência de uma camada de micro-kernel acessível livremente pela aplicação.

Resposta: d.

1. Tipos de Suporte de Tempo Real) Assinale “V” para os serviços **tipicamente** oferecidos por um micro-kernel ou “F”, caso contrário:

( ) Interface gráfica

( ) Mecanismos de comunicação entre tarefas/processos

( ) Protocolo de comunicação em rede

( ) Alocação e liberação de memória física

( ) Sistema de arquivos

Resposta: FVFVF.

1. (POSIX) Assinale a alternativa **falsa**:
   1. POSIX é um padrão criado originalmente para dar suporte a Sistemas de Tempo Real, mas que evoluiu e passou a incluir também Sistemas de Propósito Geral na sua especificação.
   2. POSIX significa “Portable Operating System Interface based on Unix” e é um padrão seguido pela maioria dos Sistemas Operacionais de Tempo Real.
   3. POSIX especifica a API de um sistema operacional, ou seja, a sintaxe das interrupções de software, que podem ser acessadas pela camada de aplicação.
   4. Exemplos do suporte oferecido pelo POSIX a Tempo Real incluem estruturas de comunicação e sincronização entre processos, como “mutexes”, variáveis-condição e filas de mensagens.
   5. Sinais permitem que processos em modo usuário possam ser informados pelo kernel acerca da ocorrência de eventos singulares e, dessa forma, sejam interrompidos para execução do “signal handler” correspondente.

Resposta: a.

1. (POSIX) Assinale a alternativa **verdadeira**:
   1. POSIX também determina aspectos temporais e de implementação dos serviços de um sistema operacional, em sua especificação.
   2. A interface POSIX de um SO geralmente apresenta melhor desempenho que a interface proprietária do mesmo.
   3. A padronização de “perfis” de POSIX auxilia na escolha de um SOTR adequado e suficiente, tendo-se em vista as necessidades do sistema sendo projetado.
   4. De acordo com a especificação POSIX, não deve haver prioridade entre as mensagens de uma fila de comunicação entre processos, que são, assim, ordenadas apenas por ordem de chegada.
   5. As tarefas ativadas por sinais emitidos pelo kernel (de acordo com o padrão POSIX) são facilmente previstas e, portanto, podem ser incluídas nos testes de escalonabilidade do sistema em tempo de projeto.

Resposta: c.

1. (Escalonamento no UNIX SVR4) Assinale a alternativa **falsa**:
   1. A solução para tempo real do SVR4, desenvolvido pela AT&T e a versão de maior sucesso comercial do System V, é muito melhor que a do UNIX tradicional.
   2. O escalonamento no SVR4 é baseado em prioridades, distribuídas inicialmente entre três classes: “time-sharing”, modo kernel e “tempo real”.
   3. Ao adotar o SVR4, o projetista da aplicação ganha a liberdade de definir novas classes de tarefas e suas prioridades, através da API deste sistema operacional.
   4. O recálculo da prioridade de uma tarefa ocorre somente após eventos associados à mesma, ao invés de a cada segundo, como em versões anteriores do System V.
   5. Uma das vantagens do SVR4 é que as tarefas da classe “tempo real” podem preemptar processos do kernel em qualquer instante, ajudando no cumprimento dos requisitos temporais do sistema.

Resposta: e.

1. Sobre as diferentes arquiteturas de sistemas operacionais, podemos afirmar que:
   1. É comum encontrar uma arquitetura baseada em “microkernel” em SOPGs.
   2. Sistemas baseados em “microkernel” tendem a ser muito mais flexíveis que sistemas baseados em “kernel” monilítico.
   3. Em um SO baseado em “kernel” monolítico, os diversos serviços fornecidos pelo SO estão organizados em camadas, todas no mesmo nível.
   4. Arquiteturas baseadas em “kernel” monolítico são ideais para aplicações de tempo real.
   5. Instalação e desinstalação dinâmica de tratadores de interrupções é uma característica indispensável em SOs baseados em “kernel” monolítico.

Resposta: b.

1. Assinale (V) ou (F) para as afirmações abaixo:
   1. ( ) Sistemas baseados em “microkernel” geralmente utilizam troca de mensagens para a comunicação dos processos.
   2. ( ) Protocolos de rede geralmente são implementados no “microkernel”.
   3. ( ) Variáveis de Condição, monitores e sinais são exemplos de serviços de sincronização de processos.
   4. ( ) Em SOs com arquitetura baseada em “microkernel”, o usuário tem a possibilidade de instalar e desinstalar módulos funcionais do SO sem a necessidade de reiniciar o sistema.
   5. ( ) Geralmente é possível encontrar um SO que apresente bons resultados para diversos sistemas com requisitos temporais críticos diferentes.

Resposta: VFVVF.

1. Associe (KM) ou (MI) para as afirmações abaixo:

- KM: “kernel” monolítico  
- MI: “microkernel”

( ) Frequente em SOs voltados para aplicações de tempo real críticas.  
( ) Arquitetura padrão do Unix.  
( ) Arquitetura dividida em camadas, onde os serviços do SO são verticalmente organizados.  
( ) Geralmente é uma arquitetura que utiliza um núcleo enxuto de funcionalidades básicas do SO.  
( ) Por possuir várias camadas organizadas verticalmente, sua previsibilidade temporal é baixa.

Resposta: MI,KM,KM,MI,KM

1. Sobre sistemas operacionais de tempo real, é incorreto afirmar que:
   1. SOPGs normalmente utilizam uma arquitetura baseada em “kernel” monolítico.
   2. SOTR normalmente possuem uma previsibilidade temporal superior à dos SOPGs.
   3. A arquitetura do QNX-RTOS é baseada em “microkernel”.
   4. SOPGs possuem uma flexibilidade maior que a dos SOTRs para tratamento de interrupções de hardware críticas.
   5. É possível encontrar soluções de tempo real soft no kernel puro do Linux, a partir da versão 2.6.

Resposta: D

1. Assinale (V) ou (F) para as afirmações abaixo:
   1. ( ) Memória compartilhada é o método, em geral, mais eficiente de implementar comunicação entre processos.
   2. ( ) “Drivers” de dispositivos geralmente são implementados no “microkernel”.
   3. ( ) Variáveis de Condição, monitores e sinais são exemplos de serviços de sincronização de processos.
   4. ( ) Em aplicações de tempo real críticas, o “kernel” deve ser totalmente preemptável, ou seja, pode ser preemptado a qualquer momento.
   5. ( ) A escolha do SO apropriado para a aplicação de tempo real depende exclusivamente dos requisitos temporais da aplicação.

Resposta: VFVVV