

1. Selecione a opção que contém o melhor conceito para segurança

- A) Característica que um sistema tem de na presença de falta não informar um estado errôneo.
- B) Característica que um sistema tem de na presença de erro não gerar uma falha.
- C) Característica que um sistema tem de na presença de falha não causar danos a vidas humanas, ao meio ambiente ou danos financeiros.
- D) Característica que um sistema tem de na ausência de falhas causar danos a vidas humanas ao meio ambiente ou danos financeiros.

2. Selecione a opção que contém o conceito correto de confiabilidade

- A) Probabilidade de um sistema operar sem causar danos ao meio ambiente,
- B) Probabilidade de um sistema operar sem causar danos a vidas humanas.
- C) Probabilidade de um sistema operar por um determinado período, em um certo ambiente, realizando uma determinada tarefa sem apresentar falhas.
- D) Probabilidade de um sistema operar por um determinado período em um certo ambiente, realizando uma determinada tarefa com a presença de muitas falhas.

3. O que é um programa multithread?

- A) Programa com duas ou mais estruturas de armazenamento
- B) Programa com duas ou mais linhas de execução
- C) Programa com dois ou mais canais de comunicação
- D) Programa com várias variáveis.

4. Cite duas dificuldades no desenvolvimento do estágio chamado de fault removal

- A) Quantidade de estados errados e falta de mão de obra para testar nas condições reais de utilização do produto serviço ou sistema desenvolvido.
- B) Quantidade de fases aleatórias e testes sem regulamentação.
- C) Quantidade de estados desconhecidos e dificuldade de testar nas condições reais de utilização do produto, serviço ou sistema desenvolvido.
- D) Quantidade de estados conhecidos e facilidade de testar nas condições reais de utilização do produto, serviço ou sistema desenvolvido.

5. O que são os termos C, P, D na tripla que representa processos periódicos ou esporádicos?

- A) C = tempo de computação D = deadline; P = período
- B) C = tempo de compilação, D = deadtime: P = período.
- C) C = tempo de encerramento; D = tempo morto ou deadtime: P = tempo de processo
- D) C = tempo de computação D = deadtime. P = tempo de resposta

6. Selecione a opção que conceitua um programa concorrente

- A) Programa com duas ou mais linhas de execução
- B) Programa com múltiplas estruturas.
- C) Programa multi-função.
- D) Programa com apenas uma linha de execução.

7. O que é um sistema de tempo real?

- A) É todo sistema que tem que responder a um estímulo de entrada em um tempo finito e especificado
- B) É todo sistema que tem que responder a um estímulo de entrada com uma resposta correta

- C) É todo sistema que tem que responder a um estímulo de entrada em um tempo ínfimo.
D) É todo sistema que tem que responder a um estímulo de entrada em um tempo de milissegundos.

8. Selecione a opção que contém a diferença de um sistema de tempo-real hard e tempo real soft.

- A) Um sistema de tempo real hard possui um deadline flexível que pode ser perdido ocasionalmente sem comprometer a operação do sistema
B) Um sistema de tempo real soft possui um deadline específico que deve ser obrigatoriamente atendido para que o sistema opere corretamente
C) Um sistema de tempo-real hard possui um deadline específico que deve ser obrigatoriamente atendido para que o sistema opere corretamente
D) Um sistema de tempo real hard possui um deadline suave que pode ou não ser atendido para que o sistema opere corretamente.

9. Quais as Finalidades dos comandos Fork e Join?

- A) O comando Fork irá permitir a concorrência de duas ou mais estruturas, e o comando Join irá permitir que o processo invocador do comando Fork aguarde pela finalização do (s) processo (s) concorrente (s).
B) O comando Fork irá permitir finalizar duas ou mais estruturas concorrentes, e o comando Join irá permitir que o processo invocador do comando Fork finalize imediatamente.
C) O comando Fork irá permitir a concorrência de duas ou mais estruturas, e o comando Join irá permitir que o processo invocador do comando Fork finalize imediatamente.
D) O comando Fork irá permitir a concorrência de duas ou mais estruturas, e o comando Join irá permitir que o processo invocador do comando Fork finalize imediatamente o processo concorrente.

10. Qual a diferença entre um processo periódico e um processo aperiódico?

- A) O processo periódico é chamado a execução a cada período de tempo, enquanto o processo aperiódico é orientado a eventos.
B) O processo periódico é chamado a execução em intervalos de tempo ínfimos enquanto o processo aperiódico é orientado a interrupções.
C) O processo periódico é chamado a execução uma única vez enquanto o processo aperiódico é orientado a eventos
D) O processo periódico é chamado a execução a cada 10s, enquanto o processo aperiódico é orientado a eventos periódicos.

11. Cite dois estágios de prevenção de faltas.

- A) fault test e fault removal
B) fault test e fault check
C) fault avoidance e fault check
D) fault avoidance e fault removal

12. Selecione o exemplo de um processo periódico

- A) Processo ou thread que pode ser invocado como uma rotina de tratamento de uma interrupção de interface serial.
B) Processo ou thread que pode ser invocado como uma rotina de tratamento de uma interrupção de timer.

- C) Processo ou thread que pode ser invocado como uma rotina de tratamento de um evento esporádico
- D) Processo ou thread que pode ser invocado como uma rotina de tratamento de uma interrupção de externa ou de porta.

13. Qual a diferença entre a falha operacional e a falha segura?

- A) Na falha operacional há a paralisação temporária da operação do sistema
- B) Na falha segura há a parada operacional do sistema**
- C) Na falha operacional há a parada operacional do sistema
- D) Na falha segura há uma operação normal do sistema

14. Cite um exemplo de uma falta intermitente.

- A) Trilha partida em uma placa de CI.
- B) Thermal Shutdown atuando em processadores.**
- C) Fonte de alimentação queimada.
- D) Solda fria na placa de CI.

15. Cite 3 tipos de faltas

- A) Permanente, perene e aleatória
- B) Aleatória, sequencial e paralela
- C) Suave, rígida e média
- D) Transiente, permanente e intermitente**

16. No escopo de código C abaixo, identifique quais são o módulo principal, o teste de aceitação e o ponto de recuperação.

```
int = x. // o valor correto de x é 5.
int i = 0;
do {
    if(i==0)
        x = f1();
    if(i== 1)
        x = f2();
    if(i==2)
        x = f3();
    i++;
}while((x!=5) && (i <=2))
```

- A) x=f1(); while((x!=5) && (i <=2)); do{**
- B) x=f1(); x=f2(); while((x!=5) && (i <=2))
- C) do{ x=f2(); x=f3()
- D) x=f1(); x=f2(); do{

17. Qual exemplo configura uma recuperação de erro para frente?

- A) int x; x = read_adc(); x = 3; ...
- B) x = read_adc(); x = 3; ...
- C) int x; x = read_adc(); if ((x<0) || x>5) x = 3; ...**
- D) int x; x = 3; int x; if(x<0) || x>5) x = 3; x = read adc(); ...

18. Qual a diferença entre a redundância dinâmica e a redundância estática?

- A) Na redundância dinâmica todos os componentes redundantes são dinâmicos.
- B) Na redundância estática todos os componentes redundantes estão inativos.
- C) Na redundância dinâmica todos os componentes redundantes estão operacionais.
- D) Na redundância estática todos os componentes redundantes estão operacionais.

19. Quando a estrutura abaixo irá finalizar?

Cobegin

F1;

F2;

F3;

Coend

- A) Somente quando todas as funções (F1, F2 e F3) finalizarem.
- B) Quando a primeira dentre as 3 funções (F1, F2 e F3) finalizar.
- C) Quando apenas uma entre as funções F1, F2 e F3 finalizar
- D) Quando duas das 3 funções finalizarem.

20. O que é um programa mono-thread?

- A) Programa com duas ou mais linhas de execução
- B) Programa com apenas uma função
- C) Programa com apenas uma estrutura de decisão
- D) Programa com apenas uma linha de execução

21. Cite duas vantagens da redundância estática

- A) Menor tempo de resposta e maior complexidade
- B) Menor tempo de resposta e maior simplicidade
- C) Maior tempo de resposta e menor simplicidade
- D) Maior tempo de resposta e maior simplicidade

22. Diferencie falta, erro e falha.

- Falta é a imperfeição física ou mecânica em um sistema de hardware ou um erro de programação em um software.
- Erro é um estado não esperado de um sistema.
- Falha é um mal funcionamento ou comportamento de um sistema.

23. Cite 3 tipos de faltas.

- Faltas transientes: uma falta transiente inicia em um tempo particular, permanece no sistema por algum tempo e então desaparece. Exemplos de tais faltas estão em componentes de hardware os quais têm uma reação adversa a alguma interferência externa, tais como campos elétricos ou eletromagnéticos. Muitas faltas em sistemas de comunicação são transientes.
- Faltas permanentes: faltas permanentes iniciam em um determinado tempo e permanecem no sistema até serem reparadas. Como exemplo, um fio partido ou um erro de projeto de software.
- Faltas intermitentes: são faltas transientes que ocorrem de tempo em tempo. Um exemplo é um componente de hardware que é sensível ao calor. Ele trabalha por um período, aquece, pára de trabalhar, volta a esfriar e então trabalha novamente (Thermal Shutdown).

24. Cite um exemplo de uma falta intermitente.

Um exemplo é um componente de hardware que é sensível ao calor. Ele trabalha por um período, aquece, para de trabalhar, volta a esfriar e então trabalha novamente (Thermal Shutdown).

25. Cite 3 tipos de falhas.

- Falha operacional (fail operational): O sistema continua a operar na presença de faltas, por um período limitado de tempo, sem perda significativa de sua funcionalidade ou desempenho;
- Falha suave (failsoft): O sistema continua a operar na presença de erros, aceitando uma degradação de sua funcionalidade ou desempenho durante a recuperação ou reparo.
- Falha segura (failsafe): O sistema mantém sua integridade enquanto aceita uma parada temporária em sua operação.

26. Diferencie redundância dinâmica de redundância estática em relação ao uso dos componentes de redundantes.

Na redundância estática, os componentes redundantes estão em constante execução com a finalidade de esconder a falta. Na redundância dinâmica, os componentes redundantes só são executados após a detecção de erro. Sendo executado assim como elemento reserva de um elemento em falta.

27. Quais as técnicas de prevenção de faltas?

Fault avoidance e fault removal.

28. Diferencie redundância dinâmica de redundância estática.

Na redundância dinâmica, apenas um componente redundante está operacional. Essa técnica possibilita a detecção de erro e sua recuperação através da execução de outro componente redundante.

Na redundância estática, todos os componentes redundantes estão operacionais, e a finalidade da técnica é mascarar o erro através de uma votação majoritária das respostas dos componentes operacionais.

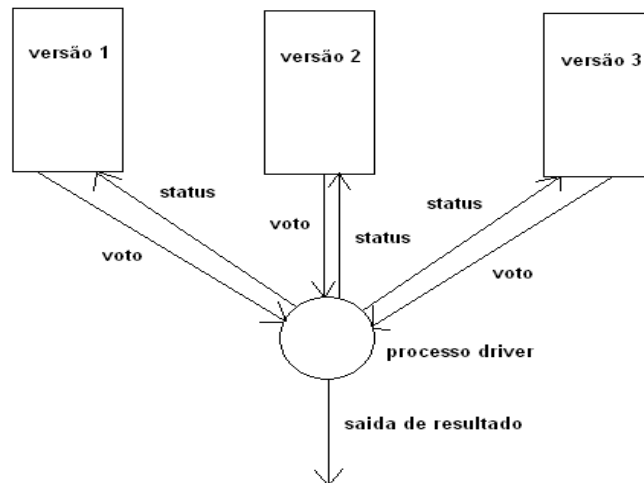
29. Cite duas técnicas de confinamento de danos.

Ação atômica e decomposição modular.

30. Cite duas técnicas de recuperação de erros.

Recuperação de erros para frente e recuperação de erros para trás

10. Apresente o diagrama de blocos de uma programação N-versão.



11. Elabore uma função de envio de mensagem assíncrona: void send_async(int *buffer, int C); e uma função de recepção: void receive(int *buffer, int C)

```
void send_async(int *buf, int c){
    canal[c] = *buf;
    return;
}
```

```
void receive(int *buf, int c){
    while(canal[c] == -1);
    *buf = canal[c];
    canal[c] = -1;
    return;
}
```

12. Elabore uma função de envio de mensagem síncrona: void send_sync(int *buffer, int C).

```
void send_sync(int *buf, int c){
    canal[c] = *buf;
    while(canal[c] != -1);
    return;
}
```