2021년 8월 14일 정보처리기사 필기 기출문제

2. UML 모델에서 한 사물의 명세가 바뀌면 다른 사물에 영향을 주며~~

해설 : 영향을 준다-> dependency

3. 익스트림 프로그래밍(XP)

- 사용자의 요구사항은 언제든지 변할수 있다.

- 고객과 직접 대면하며 요구사항을 이야기하기 위해 사용자 스토리를 활용

- 기존의 방법론에 비해 실용성 강조

4. 소프트웨어 설계에서 사용되는 추상화

- 제어 추상화 : 제어의 정확한 메커니즘을 정의하지 않고 원하는 효과를 정하는데 이용하는 방법

- 기능 추상화 : 입력 자료를 출력 자료로 변환하는 과정을 추상하는 방법

- 자료 추상화 : 자료와 자료에 적용될 수 있는 기능을 함께 정의함으로써 자료 객체를 구성하는 방법

7. 요구 분석에 설명

- 요구 분석은 소프트웨어 개발의 실제적인 첫 단계 사용자의 요구를 이해

- 요구 추출은 프로젝트 계획 단계에 정의한 문제의 범위 안에 있는 사용자의 요구를 찾음

- 도메인 분석은 요구에 대한 정보를 수집하고 배경을 분석하여 이를 토대로 모델링 진행

- 기능적 요구는 시스템이 수행해야 하는 행위들을 구체화 한 것

8. 클래스의 동작을 의미 , 클래스에 속하는 객체에 대하여 적용될 메소드 정의

- Operation

9. 분산 시스템을 위한 마스터-슬레이브 아키텍처에 대한 설명

- 실시간 시스템에서 사용

- 연산 통신 조정을 책임

- 데이터 수집기능 수행

- 마스터 프로세스는 슬레이브 프로세스를 제어가능

마스터 – 슬레이브 패턴

마스터 : 작업을 분리, 배포, 최종결과 계산 / 슬레이브 : 요청 작업 처리

10. 요구사항 정의 및 분석 설계 결과물을 표현하기 위한 다이어그램

- Data Flow Diagram

- UML Diagram

- E-R Diagram

11. 상위클래스에서 받아오는건 상속

캡슐화는 데이터를 보호하기 위해 다른 객체 접근 방지

13. GoF패턴: 23가지 디자인 패턴을 3가지 분류로 정리한 디자인 패턴

- 생성 패턴: Abstract Factory, Builder, Factory Method, Prototype, Singleton

- 구조 패턴: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy

- 행위 패턴: Chain of Responsibility, Command, Interperter, Iterator, Mediator, Memento, Observer, State, Strategty, Template Method, Visitor

14. 애자일 방법론

- 요구사항, 설계, 구현 , 시험의 단계를 통해 개발하는 방법론

- SW 개발 단계의 변화에 신속하게 대응하기 위해 요구사항을 지속적으로 분석하고 반영하여 시간 지연을 최소화 하는 방법론

- 반복적인 개발을 통한 잦은 출시 목표

- 기존 모형 문제점을 보완한 유형

- 출시 주기를 짧게 하여 다양한 요구 변화 대응

- 가볍고 실용적인 소프트웨어 개발 방법론

15. 럼바우의 객체지향 분석 기법 중 자료 흐름도를 주로 이용하는것?

럼바우 분석 기법: 객 동 기

객체모델링, 동적모델링, 기능모델링

16. 순차 다이어그램

- 객체들의 상호작용 나타내기 위해 사용

- 시간의 흐름에 따라 객체들이 주고 받는 메세지에 집중

- 동적 다이어그램

- 교류 다이어그램

18. 정해진 명령문자열을 입력하여 시스템을 조작하는 것 – CLI

19. 분산 시스템의 미들웨어?

- 복잡한 이기종 환경에서 응용 프로그램과 운영환경 간에 원만한 통신을 이룰 수 있게 해주는 소프트웨어

- 애플리케이션 – 사용자 사이 말고도 프로그램 사이도 서비스를 제공

21. 시스템 테스트 종류

- 요구사항 무결성 , 부피 , 메모리, 성능, 신뢰성, 부하, 수락, 회복, 안전, 구조, 회귀, 병행등

강도 테스트는 과다 정보량을 부과하여 정상적으로 작동하는지 검증하는 테스트

22. 버블정렬: 인접한 데이터를 비교하면서 데이터의 위치를 바꾸어 정렬하는법

24. 모듈 : 하나 또는 몇 개의 논리적인 기능을 수행하기 위한 **명령어들의 집합**

25. 테스트 드라이버: 테스트 대상 하위 모듈을 호출 하고 파라미터 전달 모듈 테스트 수행 후의 결과 도출 , 상향식 테스트에 사용됨

테스트 스텁: 제어 모듈이 호출하는 타 모듈의 기능을 단순히 수행하는 도구, 하향식 테스트

26. 선형구조: 리스트, 스택, 큐, 덱

비선형구조: 트리, 그래프

파일구조: 순차파일, 색인파일, 직접파일

27. 오버플로우 : 스택의 공간이 가득 찼는데 넣을려고 하는경우

언더 플로우 : 데이터가 없는데 꺼내려고 하는 경우

30. 검증- 소프트웨어가 요구사항에 부합하게 구현되었음을 보장하는 활동

확인: 소프트웨어가 고객의 의도에 따라 구현되었음을 보장

31.

단위 테스트: 하나의 모듈을 기준으로 진행

통합 테스트: 모듈 간

시스템 테스트: 완전한 시스템

인수 테스트: 사용자

32. 소프트웨어 형상 관리: 소프트웨어의 변경 사항을 **체계적으로 추적하**고 통제 하는 것

형상 관리 절차: 형상 식별-> 형상 통제-> 형상 감사-> 형상 기록/보고

형상 식별: 형상 관리의 **대상을 구분하고** 관리 목록의 번호를 정의하여 부여

형상 통제: 소프트웨어 형상 변경 제안을 **검토**

형상 감사: 형상 항목의 변경이 **계획에 따라 이뤄졌는지 검토/승인**

형상 기록: 소프트웨어 개발 상태에 대한 **보고서를 제공**

36. 이식성(Portability): 다양한 하드웨어 환경에서도 운용가능하게 쉽게 수정될 수 있는 능력

효율성: 최소의 작업으로 요구되는 기능을 수행하는 정도

사용용이성: 소프트웨어를 쉽게 사용 할 수 있는 정도

정확성: 사용자의 요구사항을 충족시키는 정도

37. 최악일 경우 시간복잡도

이진탐색: O(n)

AVL: O(log n)

2-3: O(log 3n)

레드 블랙 트리 : O(log n)

38. 순회

전위 : 중앙 노드 먼저

중위 : 왼쪽노드 방문 중앙

후위: 하위 모두 다음 중앙

39. 테스트 수행도구

자료 흐름도, 기능 테스트, 입력 도메인 분석, 랜덤테스트

스터브랑 드라이버는 통합 테스트 시 사용

스터브는 하향식 드라이버는 상향식

41. 무결성규칙 : 데이터무결성을 지키기 위한 모든 제약사항을 뜻함

개체 무결성: 기본키있어야함

참조 무결성: 참조관계일때 두 데이터 항상 일관된 값을 가져야함

도메인무결성: 테이블에 존재하는 필드의 무결성 보장

42. 카티션프로젝트: 두 테이블의 그저 곱집합

43. HAVING은 GROUP BY와 사용

49.슈퍼키: 유일성 만족하는 속성 또는 속성들의 집합

후보키: 유일성과 **최소성**을 만족하는 속성 또는 속성들의 집합

대체키: 기본키로 선택되지 못한 후보키

외래키: 다른 릴레이션의 기본키를 참조하는 속성 또는 속성들의 집합

51. 로킹 단위에 대한 설명

로킹 단위가 큼: 로크 수가 적어짐-> 병행성 수준이 낮아짐, 병행 제어 기법이 간단

로킹 단위가 작음 -> 로크수가 많아짐

52. 관계대수: 관계형 데이터베이스에서 정보를 검색하기 위해 어떻게 유도하는가를 보는 절차적 언어

관계해석: 비절차

54. DDL문 CREATE, ALTER, DROP / DML문 UPDATE, SELECT, INSERT

58. 타임스탬프 기법: 동시성제어를 위한 직렬화 기법으로 트랜젝션 간의 처리 순서를 미리 정의하는 방법

59. 정규형

제 1 정규형: 릴레이션에 속한 모든 도메인이 더는 분해되지 않는 **원자값으로만** 구성되어야함.

제 2 정규형: 제 1 정규형에 속하고 기본키가 아닌 모든 속성이 기본키에 **완전 함수 종속**

제 3 정규형: 제 2 정규형 + 기본키가 아닌 모든 속성이 기본키에 **이행적 함수 종속이 되지 않음**

제 4 정규형: 보이스/코드 정규형 만족 함수종속이 아닌 다치 종속 제거시 만족

제 5 정규형: 제 4정규형 + 후보키르르 통하지 않는 조인 종속 제거해야 만족

61. 응집도

시간 응집도: 같은 시간

논리적 응집도: 같은 범주에 속하는 기능끼리 묶임

우연적 응집도: 뚜렷한 관계가없음

순차적 응집도: 이전의 명령어로 부터 나온 출력결과를 그다음 명령어가 사용

62. 자동반복 요청방식 ARQ

Stop and Wait ARQ 송신 측이 하나의 블록을 전송한 후 수신측에서 에러의 발생을 점검한다음 에러발생 유무 기다리는 방식

Go back N 여러블록을 연속적으로 전송하고 수신측에서 부정 응답을 보내오면 오류이후로 부터 **다시 재전송**

Selcective-Repeat ARQ 여러 블록을 연속적으로 전송하고 수신측에서 부정응답이 오면 **발생한 블록만 재전송**

Adaptive 전송효율을 최대로 하기 위해서 데이터 블록의 길이를 채널의 상태에 따라 **동적으로 변경**

64. strcat -> 문자열을 이어주는 함수

65. Garbage Collector JAVA에서 힙에 남아있으나 더 이상 사용되지 않는 객체를 제거해주는 역할

67. JAVA 연산자 순위 🡪 -- > % > & > =

69. 몰?루

70. A class 0.0.0.0 ~ 127.255.255.255

B class 128.0.0.0 ~ 191.255.255.255

C class 192.0.0.0 ~ 223.255.255.255

D class 224~

E class 240~

72. 귀도반로섬 인터프리터, 객체지향적 -> 파이썬

74. dispatch: 준비 -> 실행

할당시간 초과: 실행->준비

대기: 실행->준비

깨움: 대기-> 준비

PCB: 운영체제가 그 **프로세스**를 관리하는데 필요한 정보를 유지하는 **자료구조 테이블**

프로세스 실행단위를 스레드

76. **순서번호는 전달하는 바이트마다** 번호가 부여

**수신번호확인** 상대편 호스트에서 받으려는 바이트의 번호를 정의

체크섬: 데이터를 포함한 세그먼트 오류를 검사한다.

윈도우 크기-> 65535byte

78. 페이지 교체 알고리즘

OPT – 앞으로 가장 오래 사용되지않을것 교체

FIFO

LRU- 오래 안사용한것

LFU – 참조횟수 낮은거

MFU – 참조횟수 가장 많은거

NUR -최근에 사용하지 않은것

80. 파일 디스크럽터

- 파일 관리를 위해 시스템이 필요로 하는 정보를 가지고있음

- 보조기억장치에 저장되어 있다가 파일ㄹ이 개방되면 주기억으로 옮겨간다.

- 파일 제어 블록

81.

이상탐지기법는 IPS(intrusion Prevent System) 예방

IDS(Instruction Dective Sysytem)는 검사하는 시스템

82. Mandatory Access Control(MAC) – 양쪽 보안 레이블에 기초하여 높은 보안 수준을 요구하는 정보가 낮은 보안주체한테 주지 않는 접근 제어방식

84. PaaS-TA: 국내 IT 서비스 경쟁력 강화를 목표로 개발되었으며 인프라 제어 및 관리 환경 실행 환경 개발 환경 서비스 환경 운영 환경으로 구성 되어있는 개방형 클라우트 컴퓨팅 플랫폼

85. DBMS에 보안 정책을 적용하는 도구인 8XDMCP를 통해 데이터 베이스에 대한 접근제어 불가능

87. VLAN- 논리적으로 LAN을 구성하여 Broadcast Domain과 구분할 수 있게 해주는 기술

89. **개인키 암호화 –블록 DES, SEED, AES, ARIA -스트림 LFSR, RC4**

공개키(비대칭) -RSA

91 . Organoc – 5만라인 이하, Semidetached – 30만라인 이하&**트렌잭션**, Embeded- 30만 라인 이하 **하드웨어 포함 실시간 시스템**

93. waterfall Model(폭포수)- 생명주기 모형중 가장 오래되고 요구사항 변경이 어려운 선형 순차적 고전적생명 주기 모형

spiral model – 대규모, 계획및 정의 – 위험분석 – 공학적개발 – 고객평가 , 위험관리 최소화 목적, **점진적으로 개발과정이 반복 되므로** 보완가능

96. MQTT: TCP/IP 기반 네트워크에서 동작하는 발행-구독 메시징 프로토콜 최근 IOT에서 많이사용됨

97. SSO: 시스템 몇대가 되어도 하나의 시스템에서 인증에 성공하면 다른 시스템에 대한 접근 권한도 얻는 시스템

98. Salt: 암호공격을 막기위해 똑같은 패스워드들이 다른 암호값으로 저장되도록 추가되는 값

99. Effort Per Task – LOC보완하며 각 기능을 구현시키는 데 필요한 노력을 생명 주기의 각 단계별로 산정 상향

전문가 감정기법 – 전문가에게 비용산정 요청 개인적이고 주관적 하향

델파이 기법- **한 명의 조정자와 여러전문가의** 의견을 종합 하향

LOC기법- **원시코드 라인수의** 비관치 낙관치 기대치 예측을 구함 상향

100. OWASP:오픈소스 웹 애플리케이션 보안 프로젝트로서 주로 웹을 통한 정보유출 등연구하는곳