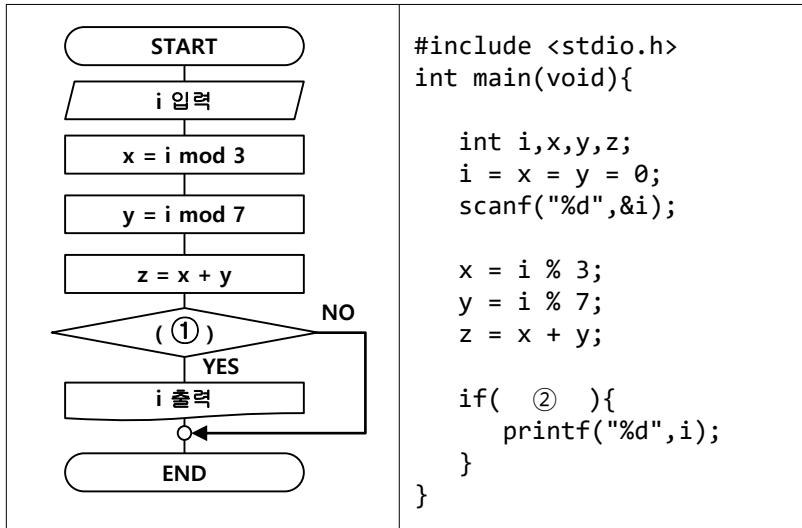

PATH

**정보처리기능사 실기
예상문제 1회**

01. 정수를 입력 받아 3의 배수이면서 7의 배수인 수를 출력하는 프로그램이 있다. 아래의 팔호 안에 적합한 표현을 작성 하시오.



02. 다음에서 설명하는 용어를 쓰시오.

숫자로 이루어져있으며, 네트워크 상에서 컴퓨터를 식별하는 고유한 주소를 IP라고 한다. 32비트 주소길이를 나타내고 있는 IPv4를 대폭 향상 시킨 차세대 주소 방식이 바로 128비트의 IPv6 이다. IPv6 전송 방식으로 (①)캐스트, (②)캐스트, (③)캐스트가 있다.

03. 다음 C 언어로 구현된 프로그램을 분석하여,
아래의 빈칸에 알맞은 값을 작성하시오.

```
#include <stdio.h>

int main(void) {
    int a, b, hap;
    a = b = 2;
    hap = ++a + b--;
    printf("hap = %d\n", hap);
    printf("a = %d이며, b = %d이다.", a, b);
}
```

04. 다음에서 설명하는 용어를 쓰시오.

(①)은 소프트웨어가 개발자가 설계한 것과 다르게 동작하거나 다른 결과가 발생되는 것이다.

에러가 원인이 되어 발생하며, 결과에서 문제를 야기할 수 있다.

예측한 결과와 실행한 결과 간의 차이점이나 업무 내용과의 불일치 등으로 인해 변경이 필요한 부분도 모두 이것에 해당한다.

- 05.** 아래 보기의 <학생> 테이블에서 학년이 2학년 이거나 3학년 학생 중 과목명이 중복되지 않게 검색하는 SQL문을 완성 하시오.

학년	성명	과목명	전화번호
1	김스오	곤충학과	010-1111-1111
2	이르반	생존학과	010-2222-2222
1	박머스	생존학과	010-3333-3333
3	강테온	신경정신과	010-4444-4444
2	유블랑	연극영화과	010-5555-5555

```
SELECT (           )
FROM 학생
WHERE 학년 IN(2,3)
```

- 06.** 단말과 단말을 이어주는, 분기와의 접합점 혹은 단말 그 자체를 노드(node)라고 한다. 이러한 노드들에 의해 자율적으로 구성되는 기반 구조 없는 네트워크를 무엇이라 하는가?

- 07.** 다음은 C언어로 구현된 25의 양의 제곱근을 구하는 프로그램이다. 빈칸에 들어갈 라이브러리 함수를 작성하시오.

```
#include<stdio.h>
#include<math.h>
int main(void) {
    double x=25;
    printf("%g", (①));
}
```

- 08.** 다음은 김백수에게 과목 테이블을 검색하는 권한을 부여하는 SQL문이다. 괄호에 들어갈 가장 적합한 명령어를 쓰시오.

```
( ) SELECT ON 과목 TO 김백수;
```

- 09.** 다음 학생 테이블에서 수학이 70점 이하인 학생이 3명 이상인 반의 튜플 수를 구하는 SQL문이다. 괄호에 들어갈 명령어를 작성하시오.

```
SELECT 반, COUNT(*) AS 학생수
FROM 학생
WHERE 수학 <= 70;
GROUP BY 반
( ) COUNT(*) >= 3;
```

- 10.** 다음 설명하는 용어를 작성하시오.

UNIX의 구조 중의 하나인 Shell은, 운영체제의 (①)과 사용자의 사이에서 인터페이스 역할을 한다. 사용자가 처음 로그인 했을 경우 (①)이 최초로 사용자에게 할당해 주는 프로그램이며, 명령어 번역기 프로그램의 일종이라 할 수 있다.

11. 다음 설명 하는 용어를 작성하시오.

프로그램 실행 여부를 확인하는 정적 테스트 방법 중 하나이다.
소프트웨어 검토를 위해 미리 준비된 자료를 바탕으로, 정해진
절차에 따라 개발자가 모집한 전문가들이 소프트웨어를 검토하는
행위를 무엇이라 하는가?

12. 다음은 JAVA 프로그래밍 오름차순 정렬에 대한 예제코드이다. 빈칸에 알맞은 값을?

```
public static void main(String[] args){  
    int ar[]={7,22,4,19,11,12};  
    int temp,i,j;  
  
    for(i=0; i<ar.length; i++){  
        for(j=0; j<ar.length-1; j++){  
            if( ①  ){  
                temp = ar[i];  
                ar[i] = ar[j];  
                ar[j] = temp;  
            }  
        }  
    }  
  
    for(i=0; i<ar.length; i++){  
        System.out.printf("%d", ar[i]);  
    }  
}
```

13.

다음 C언어 프로그래밍 예제 코드의 결과를 입력 하시오.

```
#include <stdio.h>
int main(void){
    int num1, num2, result1, result2;
    num1 = 38;
    num2 = 11;
    result1 = num1 & num2;
    result2 = num1 | num2;
    printf("result1 = %d\n", result1);
    printf("result2 = %d\n", result2);
}
```

14.

E-R Diagram은 1976년 P.Chen이 제안한 현실세계를 개념적으로 표현한 방법으로 업무 분석 결과로 도출된 개체(Entity)와 개체 관계(Relation)를 도식화한 표현이다.
다음 괄호 안에 들어가는 말로 알맞은 것은?

기호	의미
사각형	개체집합
타원	(①)
마름모	관계
선	연결

15. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.(5점)

```
#include<stdio.h>
int main(void){
    char text = 'A';
    int i = 1;

    do{
        printf("%c", text);
    }while(i<1);
}
```

16. 다음 보기에서 설명하는 것은?

키(Key)란 데이터베이스에서 조건에 만족하는 튜플(Tuple)을 찾거나 정렬할 때 기준이 되는 속성을 말한다. 그 중 릴레이션(Relation)을 구성하는 모든 튜플에 대해 유일성(Unique)은 만족하지만, 최소성(Minimality)은 만족하지 못하는 키는 무엇인가

17.

다음은 상품현황 테이블에서 이름을 기준으로 내림차순 정렬시켜서 검색하는 SQL문이다. 괄호에 들어갈 명령어를 작성하시오.

```
SELECT *  
FROM 상품현황  
ORDER BY 이름 ( );
```

18.

키보드로 명령어를 직접 입력하여 작업을 수행하는 사용자 인터페이스를 CLI라고 한다. 대표적으로 DOS, UNIX, LINUX 가 있는데 다음 기능설명에 공통적으로 들어갈 명령어를 작성하시오.

DOS	UNIX	기능
		디렉터리의 경로를 변경한다.

19. 다음 프로토콜 중 OSI 7 네트워크 계층의 프로토콜로 알맞은 것을 작성 하시오 (3개)

- HTTP
- FTP
- ARP
- UDP
- IP
- X.25
- TCP
- RS-232C

20. 다음에서 설명하는 용어를 쓰시오.(5점)

관리자가 데이터베이스 관리를 위하여 사용하는 소프트웨어로, 정의(Definition), 조작(Manipulation), 제어(Control) 기능을 가지고 있다. 사용자와 데이터베이스의 중간자 역할로, 사용자의 요구에 따라 정보를 생성해주며, 데이터베이스를 관리해주는 프로그램이다.

1.	① z = 0 ② z == 0
2.	① 멀티 ② 앤디 ③ 유니
3.	hap = 5 a = 3, b = 1
4.	결합(FAULT)
5.	DISTINCT 과목명
6.	애드록 네트워크
7.	sqrt(x)
8.	GRANT
9.	HAVING
10.	커널

11.	워크스루/검토회의
12.	ar[i] < ar[j]
13.	result1 : 2 result2 : 47
14.	속성
15.	A
16.	슈퍼키
17.	DESC
18.	cd
19.	ARP. IP. X.25
20.	DBMS

PATH 01

1. n의 배수(나머지) : [알고리즘] % 혹은 mod / [C언어] %
 - 수를 n으로 나누었을 때, 나머지 값이 0이면 그 수는 n의 배수입니다.
 ex) 5 % 2 = 1 ▶ 5는 2의 배수가 아닙니다.

2. 비교연산자(부등호) : [알고리즘] = 같다 / [C언어] == 같다

3. 요약 : 인수 i를 3과 7로 나누어 나머지가 0임을 나타내기 위하여,
 ① [알고리즘]에서는 z = 0, ② [C언어]에서는 x + y == 0

```
i = x = y = z = 0; // i : 0, x : 0, y : 0, z : 0
scanf("%d", &i); // 입력값 대입
x = i % 3; // i를 3으로 나누었을 때 나머지
y = i % 7; // i를 7으로 나누었을 때 나머지
z = x + y; // z에 x + y 대입
if(z == 0){ // 나머지가 모두 0이였을 때 인식
    printf("%d", i);
}
```

답 : ① z = 0 ② z == 0

PATH 02

구분	IPv4	IPv6
비트	32bit	128bit
할당	8bit씩 4부분	16bit씩 8부분
진법	10진수	16진수
구분기호	('.')	(':')
전송방식	유니/멀티/브로드캐스트	유니/멀티/애니캐스트
예시	192.182.0.32	fe80:0230:e8f8:7707:a5bb::a41a:1111

1. IPv6 전송 방식

유니캐스트 (1:1)

- 단일 송신자 및 수신자 통신

멀티캐스트 (1:N)

- 단일 송신자와 다중 수신자 통신

애니캐스트 (1: 가장 가까운 1)

- 단일 송신자와 가장 가까운 단일 수신자 통신

답 : ① 멀티 ② 애니 ③ 유니

PATH 03

1. 증감 연산자(+/ -)

- [++피연산자]로 사용 시 피연산자의 값을 1을 증가시키며, [--피연산자]로 1씩 감소시킵니다. 또한, 연산자의 위치에 따라 전위 연산자와 후위 연산자로 나뉘어지게 됩니다.
- 전위 연산자(++i / --i) : 피연산자 증감 후, 해당 라인 연산 수행
- 후위 연산자(i++ / i--) : 해당 라인 연산 수행 후, 피연산자 증감

```

a = b = 2;                                // a : 2, b : 2
hap = ++a + b--;                          // ++2 + 2--;
printf("hap = %d\n", hap);
    // 해당 라인(hap) 연산 실행, hap : 3 + 2-- => 5
printf("a = %d, b = %d.", a, b);      // a:3, b:1
  
```

답 : hap = 5
a = 3, b = 1

PATH 04

결함 관리 프로세스 7과정

1. 관리 계획

- 전체에 대한 계획을 수립하는 단계

2. 기록

- 발견된 결함을 관리 DB에 등록

3. 검토

- 주요 내용을 검토하고 수정할 내용을 개발자에게 전달

4. 수정

- 전달 받은 결함을 수정

5. 재확인

- 결함을 수정후 다시 테스트 수행

6. 상태 추적 및 모니터링 활동

- 관리 DB를 이용 대시보드 또는 게시판의 형태로 서비스제공

7. 최종 분석 및 보고서 작성

- 결함에 대한 정보와 이해관계자들의 의견이 반영된 보고서를 작성하고 결함 관리 종료

답 : 결함(FAULT)

PATH 05**1. 일반문**

```
SELECT 과목명  
FROM 학생  
WHERE 학년 IN(2,3)
```

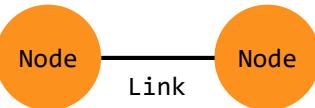
- 결과 : 학생 테이블에서 학년이 2학년이거나 3학년인 대상의 과목명들 불러옵니다(중복된 값 발생 가능).

2. DISTINCT문

```
SELECT DISTINCT 과목명  
FROM 학생  
WHERE 학년 IN(2,3)
```

- 결과 : 학생 테이블에서 학년이 2학년이거나 3학년인 대상의 과목명들을 중복 없이 불러옵니다.

답 : DISTINCT 과목명

PATH 06**1. 노드(Node)**

- 단말과 단말을 이어주는 분기와의 접합점 혹은 단말 그 자체

※ 개정 전 정보처리기능사/산업기사/기사 출제문제 경우

노드란 두 지점을 이어주는 분기 자체를 뜻하지만, 본 해설에서는 사전적 의미를 참고하여 개시하였습니다.

2. 애드혹 네트워크(Adhoc Network)

- 유선망이 고정적이지 않으면 이동 호스트로 이루어져, 유선망의 구성이 어렵거나 단기간 사용에 적합합니다.

답 : 애드혹 네트워크

PATH 07**1. 헤더(header)파일**

- 헤더(head)파일이란 특정 기능을 가진 함수의 모음으로써
ex) 우리는 '+'가 덧셈의 기능인지 알고 있지만, C언어는, '+'가 무엇인지
모르므로, 이에 어떤 기능을 수행하는지 알려줍니다.

- 제곱근(Square root)를 구하기 위해서는 해당 함수를 지원하는
헤더파일을 사용하여야 하며, <math.h>를 사용할 수 있습니다.

2. #include<math.h>

<math.h> 포함된 함수에는 대표적으로 **pow**(제곱), **sqrt**(제곱근),
ceil(올림), **floor**(내림), **abs**(절대값) 등이 존재합니다.

```
double x = 25;           // x : 25
printf("%g", sqrt(x));   // sqrt(변수) 사용 시, 5 출력
```

답 : sqrt(x)**PATH 08**

데이터베이스 언어인 SQL(Structured Query Language)는
정의어(DDL), 조작어(DML), **제어어(DCL)**로 구성되어 있습니다.

제어어 DCL(Database Control Language)는
DBMS 내에서 데이터의 보안, 무결성 등을 제어합니다.

대표적인 명령으로 COMMIT(작업완료), ROLLBACK(작업취소),
GRANT(권한부여), REVOKE(권한해제)가 있습니다. 본 문제에선 권
한을 부여한다고 했으므로 **GRANT(권한부여)**입니다.

답 : GRANT

PATH 09

1. 일반 조건문

```
SELECT 반, COUNT(*) AS 학생수
FROM 학생
WHERE 수학 <= 70;
```

- 일반적인 조건문은 **WHERE**을 사용하여 조건을 적용합니다.

2. GROUP BY를 사용할 경우의 조건문

```
SELECT 반, COUNT(*) AS 학생수
FROM 학생
WHERE 수학 <= 70;
GROUP BY 반
HAVING COUNT(*) >= 3;
```

- 그룹화(GROUP BY)를 지정하고 조건문을 적용시,
WHERE이 아닌 **HAVING**문을 사용합니다.

답 : HAVING**PATH 10**

1. 커널(Kernel)

- 쉘은 커널과 사용자의 인터페이스 역할을 수행합니다.
- 시스템 프로그램 계층이라 불리기도 합니다.
- 다중 프로그래밍 및 다중 태스킹을 위해 프로세스 관리 및 파일, 입출력 장치 등의 관리도 도맡으며 사용자나 프로그램이 처음으로 로그인 했을 경우 커널이 직접 쉘(Shell)을 사용자에게 할당합니다.

답 : 커널

PATH 11**1. 워크 스루(walk-through)**

코드의 품질을 평가하고, 개선하기 위한 목적으로 수행되는 검토 기법으로 검토회의라고도 하며 프로그램이나 설계서의 오류를 검토과정에서 발견하기 위하여 시행하는 회의 오류의 초기 검출을 목적으로 하며 발견된 오류는 문서화시키고 미리 제공된 자료들을 정해진 절차에 따라 평가합니다.

답 : 워크스루/검토회의**PATH 12****1. 반복문**

- 특정한 내용을 반복적으로 수행하기 위한 구문입니다.
 - 대표적으로 **for**, **while**, **do while**이 있습니다.
- ex) **for(기준값; 조건; 증감){ 실행문 }**

2. length 함수

- **length()**란 문자열의 길이를 표현하는 함수이며, 특정 문자 등의 배열의 길이를 뽑아내어 반복문을 이용하여 배열의 길이만큼의 값을 표시하여 반복할 수 있습니다.

3. 코드 분석

```
for(i=0; i<ar.length; i++)           // ar.length : 6, 6번 반복
    for(j=0; j<ar.length-1; j++){
        if(ar[i] < ar[j]){
            temp = ar[i];
            ar[i] = ar[j];
            ar[j] = temp;
        }
    }
}
```

답 : ar[i] < ar[j]

PATH 13**1. 비트연산자**

- 10진수의 정수 값을 2진수로 변환하여 논리 연산을 하는 기능
- 대표적으로 **&**, **|**, **^**이 있습니다.

& (and) : 값이 같은 경우 1을 반환

| (or) : 값이 둘 중에 하나라도 1이면 1을 반환

^ (xor) : 값이 다를 경우 1을 반환

2. 코드분석

```
num1 = 38;           // num1 : 38(10진수) - 0010 0110(2진수)
num2 = 11;           // num2 : 11(10진수) - 0000 1011(2진수)
result1 = num1 & num2; // 2(10진수) - 0000 0010(2진수)
result2 = num1 | num2; // 47(10진수) - 0010 1111(2진수)
```

답 : result1 : 2 result2 : 47

PATH 14

기호	의미
	개체집합(Entity)
	속성 (Attribute)
	관계(Relation)
	연결(link)

답 : 속성

PATH 15**1. 반복문**

- 특정한 내용을 반복적으로 수행하기 위한 구문입니다.
- 대표적으로 **for**, **while**, **do while**이 있습니다.

for(기준값; 조건; 증감) : 횟수 반복

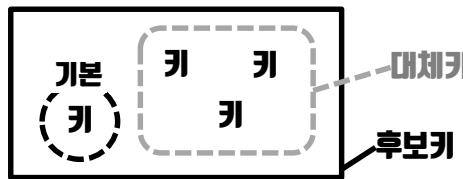
while(조건) : 조건반복

do{ 실행문 }while(조건); : 조건 부합X, 한번 반복

2. 코드분석

```
char text = 'A';      // text : A
int i = 1;           // i : 1

do{
    printf("%c", text);    // A 출력
    i++;
}while(i<1);        // 조건에 부합하지 않아도 한번 반복
```

답 : A**PATH 16****키(Key)**

1. 후보키(Key) : 유일성 / 최소성 만족

2. 기본키 : 후보키 중 선정(Not Null, 중복불가)

3. 슈퍼키 : 유일성 만족 / 최소성 불만족

- 후보 키가 없는 경우, 두 개 이상의 열을 연결하여 유일성을 만족시켜 후보 키가 되는 키

4. 대체키 : 기본 키로 선택되지 못한 후보 키

5. 외래키 : 다른 테이블의 행을 식별하는 키

답 : 슈퍼키

PATH 17**1. 오름차순(ASC)**

```
SELECT *  
FROM 상품현황  
ORDER BY 이름 ASC
```

- ASC를 사용하거나 생략 시에 오름차순 정렬이 됩니다.

2. 내림차순(DESC)

```
SELECT *  
FROM 상품현황  
ORDER BY 이름 DESC
```

답 : DESC**PATH 18****1. UNIX**

- CLI / C언어 기반 / 시분할 온라인 대화식 시스템
- 확장성, 이식성 우수

2. MS-DOS

- CLI / Windows 이전에 사용되던 운영체제

3. 명령어 비교

	UNIX	MS-DOS
경로 변경	cd	
목록보기	ls	dir
속성, 권한 변경	chmod	attrib
파일 삭제	rm	del
이름변경, 파일이동	mv	ren

답 : cd

PATH 19

1. OSI 7 계층



2. 네트워크 계층

라우팅으로 데이터를 목적지까지 빠르고 안전하게 전달하는 기능.
*경로를 선택해 패킷을 전달합니다.

3. 네트워크 계층 대표 프로토콜

a. ARP(Address Resolution Protocol)

IP 주소를 물리적인 네트워크의 주소로 대응 시키는 프로토콜

b. IP(Internet Protocol)

신뢰성 X, 연결지향 X

c. X.25

DTE(Data Terminal Equipment)와 DCE(Data Circuit-terminating Equipment) 간의 인터페이스 제공.

답 : ARP, IP, X.25

PATH 20

1. DBA(DataBase Administrator)

- 데이터베이스 관리자

2. DBMS(DataBase Management System)

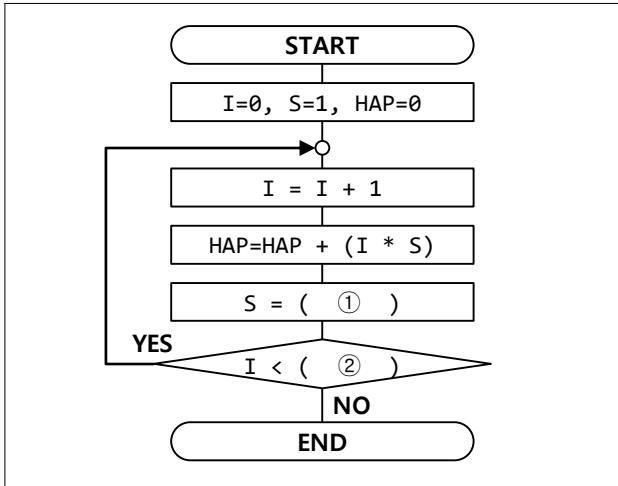
- 관리자가 데이터베이스를 관리하는 시스템/프로그램

답 : DBMS

PATH

**정보처리기능사 실기
예상문제 2회**

- 01.** 다음은 $1 - 2 + 3 \cdots + 9 - 10$ 을 계산하는 순서도이다.
괄호 안에 적합한 표현을 작성하여 순서도를 완성하시오.



- 02.** 다음은 어떠한 프로토콜에 대한 설명이다.
보기를 읽고 답을 적으시오.

- 주소 결정 프로토콜
- 논리적인 주소를 이용해 물리적인 주소를 검색
- OSI 7계층의 X.25와 같은 네트워크 계층이다.

03.

다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class function{
    public static void main (String[] args) {

        int num = 13;

        num += 1;
        num -= 2;
        num *= 3;
        num /= 4;
        num %= 5;

        System.out.printf("%d", num);
    }
}
```

04.

애플리케이션 테스트의 기본 원리 중 하나로 동일한 테스트 입력 값, 조건, 기대 결과 등으로 동일한 테스트를 반복하면 더 이상 결함이 발견되지 않는 현상을 무엇이라 하는가?

05. 다음은 프로토콜(Protocol)의 기본요소에 대한 설명이다.
빈칸에 들어갈 알맞은 말을 작성하시오.

- (①) : 데이터 형식과 부호화, 신호레벨의 규정
- (②) : 전송 조작 및 오류 제어의 정보 규정
- (③) : 신호의 순서, 속도 조정 등의 규정

06. 사이트 ‘다음’에 들어가기 위해 주소창에 ‘다음’의 주소를 입력했으나 로드된 웹페이지는 ‘네이버’ 포털 웹사이트였다. 이와 같은 상황일 때, 문제가 생긴 시스템은 무엇인가? (5점)

07. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int a, sum;
    a = sum = 0;

    while( a <= 10 ){
        a += 1;
        if(a%2 == 1)
            continue;
        sum += a;
    }

    printf("sum = %d", sum);

    return 0;
}
```

08. 다음 요청에 따라, 데이터베이스 질의를 완성하시오.

임시회원 뷰(View)를 제거하고 연관된 뷰들도 연쇄적으로 제거하시오.

```
DROP VIEW 임시회원 ( ① )
```

09.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

10.

준비 상태의 프로세스 중 가장 앞에 있던 프로세스가 CPU를 점유하게 되는 것을 의미하는 말로, 준비 상태에서 실행 상태로 변화하는 것을 지칭하는 단어는 무엇인가?

```
#include<stdio.h>
int main()
{
    int i = 0;
    int sum = 0;

    while( i <= 6 ){
        sum += i;
        i += 2;
    }

    printf("i = %d \nsum = %d", i, sum);

    return 0;
}
```

- 11.** 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class function 1{
    public static void main (String[] args){
        int n = 5;
        System.out.println(fact(n));
    }
    public static int fact(int n){
        if ( n == 1 )
            return 1;
        return n * fact(n-1);
    }
}
```

- 12.** 이백수에게 회원명단 테이블에 대해 수정(UPDATE) 할 수 있는 권한을 부여하는 명령을 완성하시오.

```
GRANT UPDATE ON 회원명단 ( ① ) 이백수
```

13. 다음이 설명하는 데이터베이스 용어는 무엇인가?

- 자료 저장공간의 최소화 및 데이터 구조의 안정성 최대화
- 데이터베이스 내부 자료의 무결성 유지 극대화
- 데이터 속성의 중복을 최소화 하여 일치성을 보장하며 데이터 모델을 단순하게 구성한다.
- (①)를 거치지 않으면 데이터베이스 내에 데이터들이 불필요하게 중복되어 예기치 못한 이상(Anomaly)현상을 야기한다.

14. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main()
{
    int result1, result2;

    result1 = 10 + 15 % 4 - 20 % 9;
    result2 = 10 * 15 % 4 - 20 % 9 + 5;

    printf("result1 = %d ", result1);
    printf("result2 = %d", result2);
    return 0;
}
```

15. 다음이 설명하는 집합 연산자는?

데이터 집합을 기준으로 다른 데이터 집합과 공통 항목을 제외한 결과만 추출한다.

16. 다음 괄호에 들어갈 알맞은 스키마 종류를 쓰시오.

스키마는 데이터베이스를 구성하는 개체, 속성, 관계 등 구조에 대한 정의와 이에 대한 제약조건 등을 기술하는 것으로 아래와 같이 3가지로 구성되어 있다.

- (①) 스키마 : 사용자나 응용 프로그래머가 각 개인의 입장에서 필요로 하는 데이터베이스의 정의
- (②) 스키마 : 데이터베이스의 전체적인 논리적 구조, 응용 프로그램이나 각각의 사용자들이 필요로 하는 데이터를 종합한 조직 전체의 데이터베이스
- (③) 스키마 : 데이터베이스의 물리적 구조로, 데이터의 실제 저장 방법을 기술한다.

17. 다음은 체육 테이블에서 운동종목이 축구인 강의실을
'B강의실'로 변경하는 SQL문을 완성하시오.

(①) 체육

SET 강의실 = 'B강의실'

WHERE 운동종목 = '축구'

18. 다음은 선수 테이블에 값을 삽입하기 위한 SQL문이다.
빈칸에 들어갈 알맞은 명령을 완성하시오.

INSERT INTO 선수(성명, 나이, 포지션)

(①) ('강규안', 23, '탱커')

19. 다음에 설명하는 애플리케이션 테스트 방법은 무엇인가?

모듈 안의 내부 구조와 작동을 직접 관찰할 수 있고, 원시 코드의 논리적인 구조를 추적하여 동작의 유효성 뿐만 아니라 실행되는 과정을 확인할 수 있는 테스트 기법

20. 다음의 보기를 보고, 설명하는 단어가 무엇인지 작성하시오.

Windows Anti-Spyware를 기반으로 한 바이러스, 스파이웨어 및 기타 형태의 맬웨어(malware)로부터 컴퓨터를 보호하도록 설계된 Microsoft의 운영 체제에 내장시킨 맬웨어 방지 도구이다. Windows Vista 및 Windows 7에선 Microsoft의 안티 바이러스 제품인 MSE(Microsoft Security Essentials)로 대체되어 사용하였으나, Microsoft는 Windows 8부터 (①)을 Windows 7 용 MSE와 매우 유사한 바이러스 백신 프로그램으로 업그레이드 하여 사용하고 있다..

- | | |
|-----|------------------------------|
| 1. | ① $S * (-1)$ ② 10 |
| 2. | ARP |
| 3. | 4 |
| 4. | 살충제 패러독스 |
| 5. | ① 구문 ② 의미 ③ 시간 |
| 6. | DNS |
| 7. | $\text{SUM} = 30$ |
| 8. | CASCADE |
| 9. | $i = 8$
$\text{Sum} = 12$ |
| 10. | 디스패치(Dispatch) |

- | | |
|-----|--------------------------|
| 11. | 120 |
| 12. | TO |
| 13. | 정규화(Normalization) |
| 14. | result1 : 11 result2 : 5 |
| 15. | MINUS |
| 16. | ① 외부 ② 개념 ③ 내부 |
| 17. | UPDATE |
| 18. | VALUES |
| 19. | 화이트박스 테스트 |
| 20. | Windows Defender |

PATH 01**1. 수의 교차적인 합과 차**

- 본 문제의 알고리즘은 반복구간을 거칠 때마다 수는 증가시키면서, 그 수의 누적 합과 차를 구하는 문제입니다.
- 수의 증가 : 인수 I가 반복문을 통해 시작값부터 종료값까지 증가

$$I = I + 1$$

◀ I에 1을 더해서 증가(반복)

- 교차적 합과 차

반복을 통해 **증가시킨 수(I)**의 **부호를 변경(S)**하여 합을 구합니다.

$$\begin{array}{l} \text{HAP=HAP} + (I * S) \\ \downarrow \\ S = (S * (-1)) \end{array}$$

◀ I에 S를 곱해서 부호 변경 후 HAP와 합하기

◀ S에 -1을 곱해서 부호 변경
S는 $(-1) \rightarrow (+1) \rightarrow (-1) \rightarrow (+1) \dots$

답 : ① $S * (-1)$ ② 10

PATH 02**1. ARP**

- 주소 재해석 및 주소 결정 프로토콜이라 불립니다.
- **논리적인(IP) 주소를 이용해 물리적인(MAC) 주소를 검색**합니다.
- 네트워크 계층에 속해있으며, 데이터 링크 계층의 MAC 주소와 연결해 줍니다.

2. RARP

- 역순 주소 결정 프로토콜이라 불립니다.
- 물리적인(MAC) 주소를 이용해 논리적인(IP) 주소를 검색합니다.
- 네트워크 계층에 속해있지만, **ARP**와는 다르게 데이터 링크 계층의 MAC 주소를 이용해 네트워크 계층의 IP를 검색해 줍니다.

답 : ARP

PATH 03**1. 산술 연산자**

- 특정 기호를 이용하여 사칙연산을 하는 작업을 의미합니다.
- [+ [덧셈]], [- [뺄셈]], *[곱셈]], /[나눗셈]], % [나머지] 등이 있습니다.

2. 복합 대입 연산자

- 산술 연산자와 대입 연산자를 간결하게 사용하는 작업입니다.

ex) 'A = A+1'를 'A += 1'로 표현

3. 코드 분석

```
int num = 13;      // num : 13
num+=1;           // n = n + 1  →  14
num-=2;           // n = n - 2  →  12
num*=3;           // n = n * 3  →  36
num/=4;           // n = n / 4  →  9
num%=5;           // n = n % 5  →  4
System.out.printf("%d", num);      // num = 4
```

답 : 4**PATH 04****1. 살충제 패러독스 (Pesticide Paradox)**

- 지속적인 살충제 살포로 내성이 생긴 벌레가 죽지 않는 현상
- 애플리케이션 테스트에서도 동일한 테스트에 대한 비정상적인 결함 검수가 이루어질 수 있습니다. 이러한 현상을 방지하기 위해서는 정기적인 리뷰와 개선이 필요하고, 다양한 시각과 방법으로 테스트하는 것이 필요합니다.

2. 테스트의 7가지 원칙

- 1) 테스트는 결함이 존재함을 밝히는 활동이다.
- 2) 완전한 테스트는 불가능
- 3) 초기 테스트
- 4) 결함 집중
- 5) **살충제 패러독스**
- 6) 테스트는 경향의 의존적이다.
- 7) 오류-부재의 궤변

답 : 살충제 패러독스

PATH 05**1. 구문(Syntax)**

- 데이터의 구조와 형식을 의미하며, 프로토콜 작성에 따라야 하는 문법과 구조입니다.

2. 의미(Semantics)

- 비트의 각 부분을 의미하며, 특정 패턴의 해석과 해석에 기초한 동작을 담당합니다.

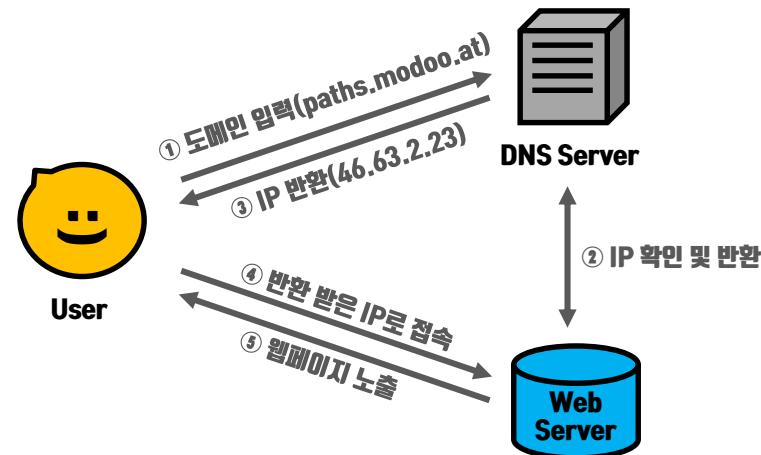
3. 시간(Timing)

- 데이터를 전송해야 할 시간의 조정과 전송속도에 대한 영역입니다.

답 : ① 구문 ② 의미 ③ 시간

PATH 06**1. DNS(Domain Name Service)**

- **IP 주소(숫자)와 도메인(문자)을 서로 변환하는 시스템.**
- DNS 서버를 이용해 자신이 입력한 도메인을 그에 할당하는 IP가 무엇인지를 검색해 불러옵니다.



답 : DNS

PATH 07

1. 반복문 제어

- 반복문에서 반복된 흐름을 벗어나기 위해 사용합니다.
- 대표적으로 **break**와 **continue**문이 있습니다.
 - * **break** : 해당 반복문을 멈추고, 탈출하는데 사용합니다.
 - * **continue** : 반복문을 유지하면서, 현재 흐름을 건너뜁니다.

2. 코드 분석

```
while( a <= 10 ){      // 반복 조건 : a가 10 이하
    a += 1;            // a + 1
    if(a%2==1)         // a가 홀수(2로 나눈 나머지가 1)일 때
        continue;       // 코드 건너뛰기 : while문 복귀
    sum += a;           // sum = sum + a
}
```

답 : SUM = 30

PATH 08

1. CASCADE : 종속

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고 있을 경우 연쇄적으로 함께 변경 또는 삭제합니다.

2. RESTRICT : 제한, 한정

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고 있을 경우 변경 또는 삭제가 취소됩니다.

답 : CASCADE

PATH 09

1. 반복문

- 특정한 내용을 반복적으로 수행하기 위한 구문입니다.
- 대표적으로 for , while, do while이 있습니다.

ex) **while(조건){ 실행문 } → 조건범위 내에서 반복 실행**

2. 코드 분석

```
while( i <= 6 ){      // 반복 조건 : i가 6 이하
    sum +=i;          // sum = sum + i (sum에 i값 누적)
    i+=2;             // i = i + 2 (i는 2씩 증가)
}

printf("i = %d\nsum = %d",i, sum);   // i = 8
                                         sum = 12
```

**답 : i = 8
sum = 12**

PATH 10

프로세스의 상태전이

1. **디스패치(Dispatch Process)** : *ready* → *running*

- 준비 상태에서 실행 상태로 바뀌는 것

2. **보류(Block Process)** : *running* → *blocked*

- 프로세스 실행 중 시간이 다 되기도 전에 입출력 동작을 해야 할 경우
프로세스가 CPU를 반납하고 보류 상태로 들어가는 것

3. **깨움(Wakeup Process)** : *blocked* → *ready*

- 보류 상태 이후 다시 준비 상태로 넘어가는 과정

4. **시간제한(Timer Process)** : *running* → *ready*

- 클락 인터럽트를 통해 프로세스가 일정 시간만 점유할 수 있게 하는 것.

답 : 디스패치(Dispatch)

PATH 11

1. 재귀 함수

- 하나의 Method(함수)에서 **자기 자신을 다시 호출**하여 해당 구문을 다시 수행하는 함수입니다.

```
fact(int n) { ←
    n = 5;
    return n * fact(n-1);
        ▶ 5 * fact(5-1)
            ↴ 4 * fact(4-1)
                ↴ 3 * fact(3-1)
                    ↴ 2 * fact(2-1)...
```

2. 코드 분석

```
public static int fact(int n){ // fact(5) 호출 시
    if ( n == 1 )           // n이 1이 될 시 호출
        return 1;             // 1의 값 반환
    return n * fact(n-1);     // 5 * 4 * 3 * 2 * 1
```

답 : 120**PATH 12**

1. GRANT : 권한 부여

ex> 이백수에게 회원 테이블 수정 권한 부여

```
GRANT UPDATE ON 회원 TO 이백수
```

2. REVOKE : 권한 해제

ex> 이백수에게 회원 테이블 수정 권한 해제

```
REVOKE UPDATE ON 회원 FROM 이백수
```

답 : TO

PATH 13**1. 데이터베이스 정규화**

- 불필요한 데이터(Data Redundancy)를 제거하는 과정입니다.
- 논리적으로 데이터 저장합니다.

2. 정규화 과정

비정규 릴레이션

1. 도메인을 분해
2. 부분적 함수 종속 제거
3. 이행적 함수 종속 제거
4. 결정자 이면서 후보키가 아닌 것 제거
5. 함수 종속이 아닌 디치 종속 제거
6. 후보키를 통하지 않은 조인 종속성 제거

1NF

2NF

3NF

BCNF

4NF

5NF

답 : 정규화(Normalization)**PATH 14****1. 산술 연산자 우선 순위**

- C언어도 수학과 동일하게 연산자에 따른 우선순위가 결정됩니다.
- 산술 연산자의 경우, 아래와 같은 우선순위를 가지게 된다.

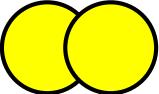
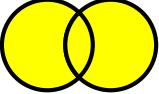
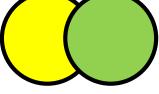
우선 순위	연산자
1	() [괄호]
2	* [곱셈], /[나눗셈], %[나머지]
3	+[덧셈], -[뺄셈]

2. 코드 분석

```
result1 = 10 + 15 % 4 - 20 % 9;
// 10 + (15 % 4) - (20 % 9)
// 10 + 3 - 2 = 11
result2 = 10 * 15 % 4 - 20 % 9 + 5;
// (10 * 15 % 4) - (20 % 9) + 5
// 2 - 2 + 5 = 5
```

답 : result1 : 11 result2 : 5

PATH 15

집합연산자	형태	정의
UNION (합집합)		중복 행이 제거된 집합
UNION ALL (합집합)		중복 행이 제거되지 않은 집합
INTERSECTIO N (교집합)		두 쿼리 결과에 공통적으로 존재하는 집합
MINUS (차집합)		첫 쿼리에 있고 두번째 쿼리에는 없는 집합

답 : MINUS

PATH 16

1. 외부스키마 = 서비스키마 = 사용자 View

- 개인(사용자나 응용프로그래머 등)의 관점
- DB에서 각 사용자가 필요로 하는 부분만 정의
- 여러 개의 외부 스키마가 존재가 가능

2. 개념스키마 = 논리스키마 = 전체적 View

- 전체 조직적인 관점(기관, 조직)
- 개체간의 관계 / 제약조건을 정의
- 하나의 개념스키마만 존재 / DBA에 의해 정의

3. 내부스키마 = 물리스키마

- 시스템프로그래머의 관점
- 물리적 저장장치의 입장에서 본 DB구조
- 개념 스키마를 기억장치에 물리적으로 구현하기 위한 방법 기술

답 : ① 외부 ② 개념 ③ 내부

PATH 17**1. UPDATE**

- 데이터베이스 조작어인 DML의 한 종류로, 데이터를 변경할 때 사용하는 명령어입니다.

2. DML

- 검색

```
SELECT column_name FROM table_name
```

- 삭제

```
DELETE column_name FROM table_name
```

- 갱신

```
UPDATE table_name
```

```
SET column1 = value1, column2 = value2...
```

- 추가

```
INSERT INTO table_name(col1,col2...)  
          VALUE (value1,value2...)
```

답 : UPDATE**PATH 18****1. INSERT INTO VALUE : 데이터 삽입**

- 새로운 데이터를 삽입할 때 사용합니다.
- 테이블 또는 뷰(View)에 한 개 이상의 행을 추가하는 명령입니다.

2. 기본형식

```
INSERT INTO table_name or view_name
```

```
VALUES value1, value2...
```

답 : VALUES

PATH 19**1. 화이트박스 테스트**

- 개발자 관점으로 내부구조와 동작을 테스트합니다.

- 종류

- 기초경로테스트 · 제어흐름테스트 · 조건테스트
- 루프테스트 · 데이터흐름테스트 · 분기테스트

2. 블랙박스 테스트

- 사용자 관점으로 명세 기반의 테스트 방법입니다.

- 종류

- 균등분할 · 한계값테스트
- 원인효과그래프테스트 · 비교테스트

답 : 화이트박스 테스트

PATH 20**1. Windows Defender**

- Windows 8 이상 운영체제에 내장된 멀웨어 방지도구

- 이전에선 MSE(Microsoft Security Essentials)를 사용하였으나, Windows 8부터는 MSE와 매우 유사한 바이러스 백신으로 업그레이드하여 **Windows Defender**를 사용하고 있습니다.

답 : Windows Defender

PATH

**정보처리기능사 실기
예상문제 3회**

01. 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class function01 {  
    public static void main (String[] args) {  
        int num = 10;  
        function(num);  
        System.out.printf("%d", num);  
    }  
  
    public static int function(int num) {  
        int sub, mod, sum;  
        sub = num - ( 10 * num / 20 );  
        mod = num % ( 10 * num / 20 );  
        sum = sub + mod;  
        return sum;  
    }  
}
```

02. 다음에서 설명하는 용어를 쓰시오.

조인(JOIN)은 2개의 테이블에 대해 연관된 튜플(Tuple)을 결합하여, 하나의 새로운 결과 집합으로 반환하는 것을 뜻한다.

조인은 크게 2가지로 구분 되는데, (①)은 일반적인 조인 형태로 교집합 형태를 하고 있으며, 관계가 설정된 두 테이블에서 조인 된 필드가 일치하는 행만을 표시한다.

03. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){

    char text[9] = {'P', 'A', 'T', 'H', 'L', 'E', 'V', 'E', 'L'};

    for(int i=8; i>=0; i--){
        printf("%c", text[i]);
    }
}
```

04. 다음에서 설명하는 용어를 쓰시오.

데이터베이스 조작 시 예기치 못한 곤란한 현상을 이상(Anomaly)이라고 한다. 이때 이상은 3가지 종류로 나누는데.

1. (①) 이상 : 불필요한 정보를 함께 저장하지 않고서는 어떤 정보도 저장할 수 없는 현상
2. (②) 이상 : 필요한 정보를 함께 삭제하지 않고서는 어떤 정보도 삭제할 수 없는 현상
3. (③) 이상 : 반복된 데이터 중에 일부를 변경 시 데이터의 불일치 또는 모순이 발생하는 현상

05. 다음에서 설명하는 용어를 쓰시오.

이 명령어는 DDL(데이터정의어) 중 하나로, 데이터베이스 관리 시스템 관리 하에 이미 존재하는 개체의 특성을 변경하거나, 기존에 존재하는 객체를 다시 정의하는 역할을 하는 언어이다.

06. 다음에서 설명하는 용어를 쓰시오.

- 키(Key)란 데이터베이스에서 조건에 만족하는 튜플을 찾거나 순서대로 정렬할 때 기준이 되는 속성을 말한다.
- 후보키 중에서 특별히 선정된 키로 중복된 값을 가질 수 없다.
- 유일성과 최소성을 가지며, 튜플을 식별하기 위해 반드시 필요한 키이다.

07. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
main() {
    int num1, num2;
    num1 = 7;
    num2 = 3;

    num1 < num2 ? ++num1 : num2--;
    printf("%d, %d", num1, num2);
}
```

08. 다음 제시된 데이터베이스 설계 순서를 완성하시오.

1. 요구조건 분석
2. 개념적 설계
3. (①)
4. (②)
5. 구현

09. 다음에서 설명하는 데이터베이스 용어를 쓰시오.
10. 다음에서 설명하는 애플리케이션 용어를 쓰시오.

사용자가 응용 프로그램과 데이터 베이스 사이에 인터페이스를 제공하고 실질적인 데이터 처리를 위해서 주로 사용하고, 데이터베이스에 저장된 자료들을 입력, 수정, 삭제, 조회하는 언어이다.

(①)은 하향식 통합 테스트에서 모듈간의 통합 테스트를 위해 일시적으로 필요한 조건만을 가지고 있으며, 실제로 동작하는 것처럼 만들어 놓은 객체이자 임시로 제공되는 시험용 모듈이다.

11. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
main()
{
    int num1, num2, result;
    num1 = 20;
    num2 = 4;

    result = num1 >> 3 << num2;

    printf("%d", result);
}
```

12. OSI 7계층 중 하나인 세션 계층을 TCP/IP 4계층으로 분류했을 때, 어느 계층에 속하는가?

- 13.** [보기]는 C언어 프로그램의 결과를 나타낸 것이다.
다음 C언어 프로그램을 완성하시오.

```
#include<stdio.h>
#define size 5
main() {
    int i, j;

    for(____①____; i<=size; i++){
        for(j=1; j<=i; j++){
            printf("*");
        }
        printf("\n");
    }
}
```

[보기]

*
**

- 14.** 다음은 1부터 500까지의 수 중에서 완전수를 구하는 C언어 프로그램이다. 빈칸에 알맞은 코드를 작성하시오.

※ 완전수: 자신을 제외한 자신의 약수를 모두 더한 값이 자신과 일치하는 수(ex. 6의 약수 1, 2, 3, 6 중 $1+2+3 = 6$)

```
#include<stdio.h>
main(){
    for(int i=1; i<=500; i++){
        int sum = 0;
        for(int j=1; j<i; j++){
            if(i%j == 0){
                sum+=j;
            }
        }
        if(sum == ____①____){
            printf("%d\n", i);
        }
    }
}
```

15. 다음에서 설명하는 용어를 쓰시오.

전체 결과의 80%가 전체 원인의 20%에서 일어나는 현상이라고 하며, 애플리케이션의 20%에 해당하는 코드에서 전체 80%의 결함이 발견된다고 하는 법칙

16. 다음의 빈칸에 들어갈 네트워크 장비를 적으시오.

(①)는 여러 개의 포트를 가진 고성능 브리지로, 기본 기능은 허브와 동일하지만, 전이중 통신방식으로 허브보다 향상된 속도를 제공한다. 또한 전송 중 패킷의 충돌이 일어나지 않도록 패킷의 목적지(MAC Address)의 지정된 포트로 직접 전송한다. (①)는 데이터 링크 계층 네트워크를 결합하여 망을 확장 구성한다.

17. 다음은 무엇에 관한 설명인가?

인공지능 기술의 하나로, 인터넷으로 연결된 사물들이 서로 데이터를 주고 받으며 스스로 분석하고 학습하여 사용자에게 정보를 제공한다. 또한 각 사물들은 인터넷에 대한 고유한 IP를 가지고 있고, 센서를 내장하고 있다.

18. 다음의 단어들과 연관된 운영체제의 명칭을 작성하시오.

- | | | |
|-----------|----------|---------|
| · 멀티 태스킹 | · 다중 사용자 | · C언어기반 |
| · 시분할운영체제 | · 트리구조 | · 커널 |
| · 쉘 | · 서버 특화 | · 오픈소스 |

19. 다음에서 설명하는 용어를 쓰시오.

20. MS-DOS의 명령어 중 하나이면서, 하위 디렉터리를 포함해 파일 및 디렉터리를 전부 복사하는 명령어는 무엇인가?

입력, 실행 조건, 테스트 절차 및 특정 프로그램 경로를 실행하거나 검증하는 것과 같은 특정 소프트웨어 테스트 목표를 달성하기 위해 실행될 단일 테스트를 정의하는 테스트 항목에 대한 명세서

1.	10
2.	INNER JOIN
3.	LEVELHTAP
4.	① 삽입 ② 삭제 ③ 갱신
5.	ALTER
6.	기본키(Primary Key)
7.	7, 2
8.	① 논리적 설계 ② 물리적 설계
9.	DML
10.	스텝(테스트 스텝)

11.	32
12.	응용 계층
13.	i = 1
14.	I
15.	파레토 법칙
16.	스위치
17.	IOT(Internet of Things)
18.	리눅스(Linux)
19.	테스트 케이스
20.	Xcopy

PATH 01**1. 메소드(Method)**

- 반복적인 코드의 사용을 방지하기 위하여, 하나의 항을 만들어서 필요할 때마다 해당 값을 호출하여 반환하는 역할을 수행합니다.

2. 코드 분석**- 메인구문**

```
public static void main (String[] args) {
    int num = 10;      // num : 10
    function(num);     // 저장 공간 X, 출력 불가
    System.out.printf("%d", num); }      // 10
```

- 메소드

```
public static int function(int num){
    int sub, mod, sum;
    sub = num - ( 10 * num / 20 );      // 5
    mod = num % ( 10 * num / 20 );      // 0
    sum = sub + mod;                  // sum : 20
    return sum; }                     // sum 반환
```

function(10)

답 : 10**PATH 02****1. INNER JOIN(내부 조인)**

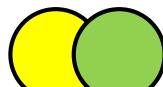
- 두 테이블에서 조인된 필드가 일치하는 행만을 표시



INNER JOIN

2. OUTER JOIN(외부조인)

- 조인 조건에 만족하지 않는 튜플도 결과로 출력하기 위한 조인

LEFT
OUTER JOINRIGHT
OUTER JOINFULL
OUTER JOIN**답 : INNER JOIN**

PATH 03

1. 배열(Array)

- 같은 자료형을 가진 변수들의 집합입니다.

Array [index_number] = { Value1, Value2... }

▶ 0부터 시작

- 배열 선언 시,

변수에 대한 사용 개수를 입력하며 ▶ int num[2] = { 1, 2 };
0을 기준으로 출력됩니다. ▶ num[0] → 1, num[1] → 2

```
char text[9] = {'P', 'A', 'T', 'H', 'L', 'E', 'V', 'E', 'L'};  
// text(배열)의 변수의 개수는 9개 선언  
for(int i=8; i>=0; i--){ // 8부터 0까지 반복  
    printf("%c", text[i]); // text[8부터 0까지] 출력  
}
```

답 : LEVELHTAP

PATH 04

1. 이상(Anomaly)

- 데이터 중복성에 의해서, 릴레이션 조작 시 예기치 못한 곤란한 현상 또는 데이터 불일치 현상입니다.

- 종류 : 삽입이상 / 삭제이상 / 갱신이상

- 이상을 해결하기 위해서는 정규화(Normalization)을 실행하여, 테이블 내의 데이터 중복을 제거합니다.

답 : ① 삽입 ② 삭제 ③ 갱신

PATH 05

1. DDL(Data Definition Language)

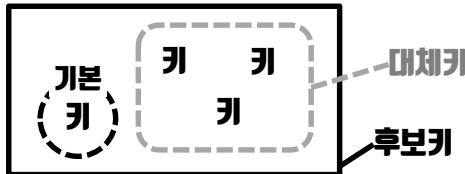
- 데이터베이스 구조를 정의하는 언어이며,
 데이터를 생성(CREATE), 수정(ALTER), 삭제(DROP)하는 등
 데이터 전체의 구조를 결정하는 역할을 하는 언어입니다.

종류	구분	역할
CREATE	생성	객체 생성
ALTER	수정	객체 수정
DROP		객체 삭제
TRUNCATE	삭제	객체 내용 삭제 (초기화)

답 : ALTER

PATH 06

키(Key)



1. 후보키(Key) : 유일성 / 최소성 만족
2. 기본키 : 후보키 중 선정된 키로 중복 값 입력이 불가능하다.
Null 값을 가질 수 없다.
3. 슈퍼키 : 유일성 만족 / 최소성 불만족
4. 대체키 : 기본 키로 선택되지 못한 후보 키
5. 외래키 : 다른 테이블의 행을 식별하는 키

답 : 기본키(Primary Key)

PATH 07**1. 삼항 연산자**

- 조건(논리연산)에 부합할 경우, True와 False에 해당하는 값을 출력하는 구문을 의미합니다.
 - 형식 = 조건 ? True : False
- ex) 10 > 3 ? 'A' : 'B' ▶ 'A' 출력

2. 증감 연산자

- 피연산자를 1씩 증가(혹은 감소)시켜 주는 구문을 의미합니다.
 - 위치에 따라 전위와 후위로 나누게 된다.
- ex) ++피연산자 : 1의 값 증가 이후, 해당 라인 연산
피연산자-- : 해당 라인 연산 이후, 1의 값 감소

3. 코드 분석

```

num1 = 7;    // num1 : 7
num2 = 3;    // num2 : 3
num1 < num2 ? ++num1 : num2--;
// 7 < 3(false) ? ++7 : 3-- → 3--
printf("%d, %d", num1, num2); // 7, 2
// 조건에 부합되지 않아서, True값을 인식하지 않습니다.

```

답 : 7, 2**PATH 08****1. 데이터베이스 설계**

- 사용자의 요구를 분석하고, 이를 데이터베이스의 구조에 맞게 변형한 후, 구현하여 일반 사용자들에게 제공하기 위한 과정입니다.

요구조건분석

요구 조건에 대한 명세 작성

개념적 설계

모델링 작업 / E-R모델링 / 개념스키마

논리적 설계

인터페이스 작업 / 논리적 스키마

물리적 설계

물리적 구조 변환

구현

데이터베이스 생성

답 : ① 논리적 설계 ② 물리적 설계

PATH 09

1. DDL(정의어; Data Definition Language)

- 데이터베이스 구조, 데이터형식, 접근 방식 등 데이터베이스를 구축하거나 수정할 목적으로 사용하는 언어입니다.

2. DML(조작어; Data Manipulation Language)

- 데이터베이스 사용자와 데이터 관리 시스템간의 인터페이스를 제공 프로그램이나 질의어를 통하여 저장된 데이터를 조작하여 허용되는 언어입니다.

3. DCL(제어어; Data Control Language)

- 데이터의 보안, 무결성, 권한, 회복, 병행 제어 등을 정의하는데 사용하는 언어이다. 주로 데이터 관리를 목적으로 사용합니다.

답 : DML

PATH 10

구분	스텝	드라이버
시기	상위 모듈은 있지만 하위 모듈이 없을 때, 하위 모듈 로의 테스트를 진행	상위 모듈 없이 하위 모듈만 있을 때, 상위 모듈 로의 테스트를 진행
방식	하향식 테스트	상향식 테스트
특징	작성이 쉬운 시험용 모듈	존재 하지 않는 상위 모듈간의 인터페이스 역할

답 : 스텝(테스트 스텝)

PATH 11**1. 시프트 연산자**

- 10진수를 2진수로 변환하여 자리를 이동하는 기능을 의미합니다.

- 대표적으로 `>>`, `<<` 등이 있습니다.

`>>` 값 : 숫자만큼 비트 위치를 오른쪽으로 이동

`<<` 값 : 숫자만큼 비트 위치를 왼쪽으로 이동

ex) num = 4; // 0000 0100

num << 2 // 0000 0100 << 2

결과 : 16 // 0001 0000(2진수) → 16(10진수)

2. 코드분석

```
num1 = 20; // num1 : 20(10진수) → 0001 0100(2진수)
num2 = 4; // num2 : 4
```

```
result = num1 >> 3 << num2;
```

```
// 0001 0100 >> 3
```

```
// 0000 0010 << 4
```

```
// 0010 0000(2진수) → 32(10진수)
```

답 : 32**PATH 12****OSI 7 계층**

응용(Application)

표현(Presentation)

세션(Session)

전송(Transport)

네트워크(Network)

데이터링크(Data Link)

물리(Physical)

TCP/IP 4 계층응용계층
(Application)

전송계층(Transport)

인터넷(Internet)

네트워크 액세스
(Network Access)**답 : 응용 계층**

PATH 13

1. 반복문

- 특정한 내용을 반복적으로 수행하기 위한 구문입니다.
- 대표적으로 **for**, **while**, **do while**이 있습니다.

ex) **for(기준값; 조건; 증감){ 실행문 }**

2. 코드 해석

```
#define size 5      //size : 5
main() {
    int i, j;

    for(i=1; i<=size; i++){      // size의 값만큼 반복
        for(j=1; j<=i; j++){    // i의 값만큼 반복
            printf("*");
            // 반복1회 *, 반복2회 ** ...
        }
        printf("\n");           // 내부 for문 끝나면 출력
    }
}
```

답 : i = 1

PATH 14

1. 완전수(미지수 n의 완전수 여부확인)

- a. 미지수 n의 약수를 구하여 그 합을 구한다($n \% 1, n \% 2 \dots n \% n$).

- 반복문을 사용하여, 미지수 n을 1부터 n까지의 수로 나눈 나머지를 확인하고 그 합을 구한다. **(나머지가 0이면, 그 수는 n의 약수이다.)**

- b. 약수의 합을 미지수 n과 비교한다.

2. 코드 해석

```
for(int i=1; i<=500; i++){          // i는 1~500까지 반복
    int sum = 0;                      // 합계 변수 선언
    for(int j=1; j<i; j++){          // j는 1~(i-1)까지 반복
        if(i%j == 0){                // 약수를 확인하는 if문
            sum+=j;                  // 약수와 자기 자신을 대입
        }
    }
    if(sum == i){                    // 합계가 i 2와 같을때
        printf("%d\n", i);
    }
}
```

답 : i

PATH 15**1. 파레토 법칙**

- 빌프레도 파테로가 '이탈리아 인구의 20%가 이탈리아 전체 부의 80%를 가지고 있다'는 발언에서 시작한 법칙
- 현대에 들어서 애플리케이션 테스트에도 적용중인 법칙
- 발견된 80%의 오류는 20% 모듈에서 발견되므로, 20%의 모듈을 집중적으로 테스트합니다.

답 : 파레토 법칙**PATH 16****1. 리피터**

- 네트워크 전송 신호를 증폭한 후, 반환 시켜주는 장치로, 물리층 기능을 수행하며 네트워크 망의 물리적인 길이를 확장합니다.

2. 허브

- 하나의 대역폭을 분배하는 장치로, 네트워크 망의 물리적인 길이를 확장하는데 사용합니다. 물리적으로 성형 구조, 논리적으로는 버스형 구조의 물리층 기능을 수행합니다.

3. 브리지

- 같은 구조의 네트워크를 연결하는 장치로, 물리/데이터 링크의 기능을 수행하며 큰 네트워크를 작은 세그먼트로 분할합니다.

4. 라우터

- 네트워크 상의 최적의 경로를 제공하며, 물리/데이터 링크/네트워크 층의 기능을 수행합니다. 또한 다중상호연결망에서 패킷을 중계합니다.

5. 게이트웨이

- 서로 다른 구조나 프로토콜을 가진 네트워크를 연결하며, 모든 계층의 기능 수행하는 프로토콜 변환기입니다.

답 : 스위치

PATH 17**1. IOT(Internet of Things; 사물인터넷)**

- 사물인터넷이란 **RIFD**와 같은 센서를 이용해 사물에 탑재한다.
자칫하면 유비쿼터스랑 혼갈릴 수 있지만 자체적인 통신이 아닌
인터넷을 포함한 사물이므로 확장된 개념이라 생각하면 편하다.
- 사용분야 : 웨어러블 디바이스, 자율주행 차, 스마트홈

답 : IOT(Internet of Things)

PATH 18**1. Unix**

- C언어 기반의 운영체제로, 1960년대 MIT 대학과 벨 연구소에서 개발 되었습니다.
- 다중 사용자 / 멀티태스킹, 시분할 방식을 사용하고 있습니다.
- 커널(Kernel), 쉘(Shell), 유틸리티 프로그램이 있습니다.

2. Linux

- Unix에서 유래된 **오픈 소스** 운영체제입니다.
- 특징으로 다중 사용자, 멀티태스킹, 다중 스레드 지원이 있으며, 높은 보안성과 안전성을 가지고 있는 네트워크 운영 체제입니다.

※ Android OS는 유닉스 및 리눅스를 기반으로 작성되었습니다.

답 : 리눅스(Linux)

PATH 19**1. 테스트 케이스**

- 특정 요구사항 준수 여부를 확인하기 위해 개발된 입력 값, 실행 조건, 예상된 결과의 집합, 미리 설계하면 오류를 방지할 수 있고 테스트 수행에 필요한 인력, 시간 등의 낭비를 줄일 수 있습니다.

*** 테스트 케이스 작성 절차**

계획 검토 및 참조 문서 수집

내부 검토 및 우선순위 결정

요구사항 정의

테스트 설계와 방법 결정

테스트 케이스 정의

테스트 케이스 타당성 확인 및 유지보수

테스트 수행

답 : 테스트 케이스**PATH 20****1. xcopy(구조복사)**

- 서브 디렉터리를 포함한 파일과 디렉터리를 복사할 수 있습니다.
- 매개 변수를 사용하면 백업 장소까지 한 번에 정할 수 있다.

ex> xcopy "C:\Temp" "F:\PATH" /h /e /y

의미 : Temp 안의 하위 디렉터리 및 파일을 복사한 후,
전부 PATH에 붙여 넣는다.

매개변수 : /h (숨김 파일 포함), /e (빈 폴더 포함), /y (모든 질문 없이 복사)

답 : xcopy

PATH

**정보처리기능사 실기
예상문제 4회**

01. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
#include<stdlib.h>

int function(int num1[], int num2[]){
    int *num;
    int i, j, temp;
    num = malloc(sizeof(int)*6);

    for(i=0, j=3; i<3, j<6; i++, j++){
        num[i] = num1[i];
        num[j] = num2[i]; }

    for(int i =0; i<5; i++)
        for(int j=0; j<i; j++)
            if(num[i] <= num[j]){
                temp = num[i];
                num[i] = num[j];
                num[j] = temp;
            }

    return num;
    free(num);
}
```

02. 네트워크를 통해 외부 컴퓨터와의 파일 생성 및 전송 등에 사용하는 원격 서비스로, 암호화 기능을 지원하지 않아서 보안 문제가 발생할 수 있는 프로토콜은 무엇인가?

```
int main(){
    int A[] = {12, 23, 10};
    int B[] = {18, 32, 52};
    int *C =
        malloc(sizeof(int)*6);

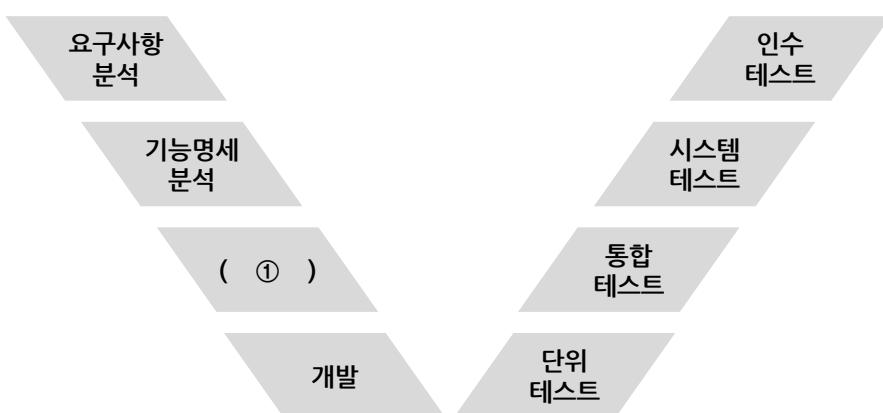
    C = function(A, B);

    for(int i=0; i<6; i++)
        printf("%d ", C[i]);
    free(C);
}
```

- 03.** 다음 SQL문은 학원 테이블에서 수강과목이 PC인 학생들의 이름과 시간을 '종합표'라는 가상의 테이블로 작성하는 식이다. 빈칸에 들어갈 알맞은 단어를 작성하시오.

```
CREATE (      ) 종합표(이름, 시간) AS  
SELECT 이름 시간  
WHERE 수강과목 = 'PC';
```

- 04.** 다음 소프트웨어 생명 주기의 V-모델의 빈칸으로 알맞은 것은?



05. 다음은 C언어로 구현된 두 수의 차이를 구하는 프로그램이다. 아래의 빈칸에 맞는 라이브러리 함수를 입력하시오.

```
##include<stdio.h>
#include<stdlib.h>
void main(){
    int result;
    result = function(20, -31);

    printf("두 수의 차이는 : ");
    printf("%d",result);
}

int function(int i, int j){
    int result;
    result = ( ① )(i) - ( ① )(j);

    return result;
}
```

06. 다음에서 설명하는 프로토콜은 무엇인가?

(①)는 데이터 전송에 사용하는 프로토콜로, 데이터의 경계가 존재하여 송수신 사이에 전송하는 데이터를 똑같이 크기로 배분하여 읽는다. 또한 (①)는 데이터 전송시 오류가 발생하면 재전송을 하지 않기에 신뢰성이 낮다.

07. IPv4 주소의 부족으로 인하여, 기존의 네트워크를 최대한 활용하기 위해 사용 목적에 따라 각 규모에 따른 네트워크 주소와 호스트 주소를 분류하여 관리를 용이하게 한 것은?

08. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
struct function{
    int num1;
    int num2;
};

void main(){
    struct function v1; struct function v2;
    v1.num1 = 19; v1.num2 = 42;
    v2.num1 = 30; v2.num2 = 12;

    if(v1.num1 < v1.num2)
        printf("%d ", v1.num1 & v1.num2 | v2.num1);
    if(v2.num1 > v2.num2)
        printf("%d ", v2.num1 | v2.num2 & v1.num1);
}
```

09. SQL문 내의 ALTER TABLE과 관련된 컬럼 명령어이다.
다음의 빈칸에 들어갈 단어는?

- (①) - 테이블에 필요한 컬럼을 추가
- DROP COLUMN - 테이블에 필요 없는 컬럼을 삭제
- MODIFY COLUMN - 테이블에 대한 제약조건을 변경
- RENAME COLUMN - 테이블 생성 시 만든 컬럼명을 변경

10. 다음은 <시험> 테이블에서 점수가 80~90점인 사람의 이름
을 출력하는 구문이다. 빈칸에 들어갈 단어를 작성하시오.

```
SELECT 이름
FROM 시험
WHERE 점수 ( ① ) 80 ( ② ) 90;
```

11. 다음의 설명하는 정의로 알맞은 것은?

- 프로그램과 명세서 간의 차이, 업무 내용 불일치이다.
- 기대 결과와 실제 관찰 결과 간의 차이이다.
- 시스템이 사용자가 기대하는 타당한 기대치를 만족시키지 못할 때 변경이 필요한 모든 것을 의미한다.

12. 다음의 빈칸에 들어갈 말을 작성하시오.

- (①)은 오라클 사가 관리/배포하는, 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)이다.
- (①)은 유닉스나 리눅스, 윈도우 운영 체제 등에서 사용할 수 있으며, 다중 사용자, 다중 스레드를 지원한다.

13. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int temp;
    int num[] = {1, 23, 42, 19, 50};

    for(int i=0; i<4; i++)
        for(int j=0; j<4; j++)
            if(num[i] >= num[j]){
                temp = num[i];
                num[i] = num[j];
                num[j] = temp;
            }
    for(int i=0; i<sizeof(num)/sizeof(int); i++)
        printf("%d ",num[i]);
}
```

14. 다음은 시험 테이블에서 과목이 수학인 학생들의 이름과 점수를 추출하는 연산문과 결과로, 관계 대수의 일종이다.
빈칸에 들어갈 알맞은 단어를 작성하시오.

이름	과목	점수	총점
김재연	수학	92	A
강민준	과학	80	D
나시찬	수학	84	C
장지웅	국어	87	B



이름	점수
김재연	92
나시찬	84

$\Pi(\textcircled{1}) (\Sigma(\textcircled{2})(시험))$

15. 다음이 설명하는 단어는 무엇인가?

UNIX 내에서 소유자, 허가내용, 파일 등이 변경될 때 같이 변경되는 파일 시스템 및 블록의 종류다. 각 디렉토리와 파일의 대한 대부분의 정보를 가지고 있는 레코드이기도 한다. 삭제라는 개념은 없으나 파일의 정보를 담고 있기 때문에 그에 대한 연결을 해지하는 개념으로 본다.

16. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int num;
    num = 1;

    switch(num){
        case '1':
            printf("%d", num + 2); break;
        case '2':
            printf("%d", num + 1); break;
        case '3':
            printf("%d", num + 3); break;
        default:
            printf("%d", num); break;
    }
}
```

17. 다음은 DB 트랜잭션의 특성인 ACID에 대한 설명이다.
빈칸에 들어갈 알맞은 단어를 작성하시오.

원자성(Atomicity) : 실행 중 중단되지 않게 하는 능력

일관성(Consistency) : 실행의 완료 후 데이터베이스 상태를 유지

격리(Isolation) : 한 트랜잭션의 실행 중 다른 트랜잭션이
끼어들지 못하게 보장

(①)(Durability) : 실행에 성공한 트랜잭션은 영원히 반영

18. 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include<stdio.h>
int function(int num);

void main(){
    int num=5;
    num = function(num);
    num++;
    printf("%d", num);
}

int function(int num){
    ++num;
    return num++;
}
```

19. 다음에서 설명하는 Windows 단축키를 쓰시오.

Windows 10의 바탕화면보기 단축키로, 실행 중인 모든 창을 최소화 시켜 바탕화면을 보여준다. 다시 한 번 눌렀을 경우 내려간 창들이 다시 원위치로 돌아온다.

20. 다음의 지문이 설명하는 단어는 무엇인가?

(①)은 주로 서버 중심의 동작으로 소프트웨어들의 논리적 형상의 구조를 설명하며, 소프트웨어 컴포넌트들과 관계를 정의한 시스템 구조이다.

컴포넌트들 사이의 상호관계를 설계하며 전개하기 위한 지침 및 원리이고, 간략성 및 추상화 등의 특징 또한 가지고 있다.

1.	10 12 18 23 32 52
2.	TELNET
3.	VIEW
4.	설계(Design)
5.	Abs
6.	UDP
7.	네트워크 클래스
8.	30 30
9.	ADD
10.	① BETWEEN ② AND

11.	결함(Defect)
12.	MySQL
13.	50 42 23 19 1
14.	① 이름, 점수 ② 과목='수학'
15.	i-node
16.	1
17.	지속성(영속성)
18.	7
19.	윈도우키 + D
20.	소프트웨어 아키텍처 (Software Architecture)

PATH 01

```
#include<stdio.h> // 입력, 출력 관련 헤더 파일
#include<stdlib.h> // 난수 및 동적 메모리 할당 등 사용
int function(int num1[], int num2[]){ // 서브 함수 생성
    int *num; // 포인터 변수
    int i, j, temp;
    num = malloc(sizeof(int)*6); // 동적 메모리 할당

    for(i=0, j=3; i<3, j<6; i++, j++){ // 두 가지 변수를 반복
        num[i] = num1[i]; // num[0~2] = num1[0~2] 각각 대입
        num[j] = num2[i]; // num[3~5] = num2[0~2] 각각 대입
        // num => 12, 23, 10, 18, 32, 52
    }
    for(int i=0; i<5; i++) // 배열의 값 확인 용도로 사용
        for(int j=0; j<i; j++) // 나머지 배열의 값으로 비교 용도
            if(num[i] <= num[j]){ // 각 배열의 값을 비교
                temp = num[i];
                num[i] = num[j];
                num[j] = temp; }
    // temp 변수를 이용하여, num[i]의 값과 num[j] 스왑

    return num; // 포인터변수 NUM 반환
    free(num); // 동적메모리 할당 해제}

```

답 : 10 12 18 23 32 52

PATH 02**1. TELNET**

- 네트워크 기반의 원격 통신 프로토콜
- 데이터 전송 시 **암호화 작업이 없어서**, 보안상 위험성이 높습니다.

2. SSH(Secure Shell)

- 원격 통신에 이용되는 프로토콜
- 데이터 전송 시 **암호화 기법을 통해**, 보안상의 위험성이 낮습니다.

3. FTP(File Transfer Protocol)

- 파일 등의 데이터들을 전송 및 교환에 사용되는 프로토콜.
 - FTP 서버에 접속하기 위해, 계정과 암호가 필요합니다.
- ※ Anonymous FTP는 익명성을 보장합니다.

답 : TELNET

PATH 03

1. 뷰(View)

- 하나 이상의 기본 테이블로부터 유도된 **가상의 테이블**을 입니다.
- 구조와 조작도 기본 테이블과 매우 유사합니다.

2. 뷰 생성

- 대표적인 구문의 형식은 아래와 같습니다.

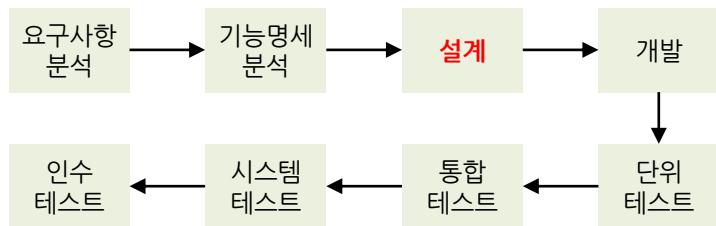
```
CREATE VIEW 뷰이름 AS  
SELECT 컬럼1, 컬럼2, ...  
FROM 테이블  
WHERE 조건문;
```

답 : VIEW

PATH 04

1. V-모델

- 소프트웨어 개발 프로세스인 폭포수 모델의 확장된 형태
- 폭포수 모델과 달리, 왼쪽은 아래로 오른쪽은 위로 진행하며 알파벳 V모양으로 진행 된다.
- 순서



답 : 설계(Design)

PATH 05**1. 헤더 파일 <stdlib.h>**

- 동적 메모리 할당, 난수 등의 함수를 불러오는데 사용합니다.

ex) `rand()` : 0 ~ 32767까지의 난수 생성 (코딩 재실행 시초기화 X)

`srand()` : 무작위 난수 생성 (코딩 재실행 시초기화 ○)

`malloc()` : 동적 메모리 할당 (공간을 유동적으로 변경 가능)

`abs()` : 입력한 인수를 **부호 없는 숫자로 변경 (절대값)**

답 : abs**PATH 06****1. UDP (User Datagram Protocol)**

- **흐름제어를 하지 않기 때문에 TCP보다 데이터 전송속도가 빠르다.**

- TCP의 경우 한번에 모든 데이터를 읽을 수 있지만,

UDP는 여러 번의 데이터를 전송했다면, 여러 번을 읽어야 합니다.

(데이터 경계가 존재)

- 송수신 중 데이터 누락/오류 발생 시, **재전송하지 않으며**,

데이터의 순서를 유지하지 않는다.(비신뢰성 프로토콜)

- 연결 설정이 없는 **비연결성 프로토콜**이다.(1vs1, 1vs다수 등)

답 : UDP

PATH 07

1. 네트워크 클래스(Classful network)

- 네트워크 단말의 증가로 가용 가능한 IPv4의 주소가 부족해졌고,

이에 **사용 목적에 따라 IP의 대역대를 나누어** 각 규모에 따라

관리하기 쉽게 표현한 것을 “클래스”라 합니다.

Class	IP주소의 첫째 옥텟	사용 목적	이론적 IP 주소 범위
A class	0xxx xxxx	대륙 안통신	0.0.0~ 127.255.255.255
B class	10xx xxxx	국가안통신	128.0.0 ~ 191.255.255.255
C class	110x xxxx	기업안통신	192.0.0 ~ 223.255.255.255
D class	1110 xxxx	그룹통신	224.0.0 ~ 239.255.255.255
E class	1111 xxxx	연구용통신	240.0.0 ~ 255.255.255.255

답 : 네트워크 클래스

PATH 08

1. 구조체

- 변수 선언의 경우, 기존에는 필요할 때마다 새로 선언을 하지만, **<구조체>**의 경우 **하나 이상의 변수를 그룹화하여 사용합니다.**

// 구조체의 변수를 호출명을 통해 하나 이상을 호출 가능

2. 코드 분석

```
v1.num1 = 19; v1.num2 = 42; // 0001 0011, 0010 1010
v2.num1 = 30; v2.num2 = 12; // 0001 1110, 0000 1100

if(v1.num1<v1.num2) // v1.num1보다 v1.num2가 초과시 인식
    printf("%d ", v1.num1 & v1.num2 | v2.num1);
    // 0001 0011 &(and) 0010 1010 => 0000 0010 (2)
    // 0000 0010 |(or) 0001 1110 => 0001 1110 (30)
if(v2.num1>v2.num2) // v2.num1| v2.num2보다 초과시 인식
    printf("%d ", v2.num1 | v2.num2 & v1.num1);
    // 30 (연산자 우선순위 : &(and) -> ^xor) -> |(or)순
```

답 : 30 30

PATH 09

1. ALTER TABLE

- ADD : 컬럼 추가

```
ALTER TABLE 테이블명  
ADD 컬럼명 데이터 유형;
```

- DROP COLUMN : 컬럼 삭제

```
ALTER TABLE 테이블명  
DROP COLUMN 컬럼명;
```

- MODIFY COLUMN : 제약조건 변경

```
ALTER TABLE 테이블명  
MODIFY 컬럼명 데이터 유형;
```

- RENAME COLUMN : 컬럼명 변경

```
ALTER TABLE 테이블명  
RENAME COLUMN 컬럼명 TO 새로운컬럼명
```

답 : ADD

PATH 10

1. BETWEEN A AND B

- SQL 구문의 조건문에서 지정한 **두 수 사이의 숫자**를 반환합니다.

ex) 나이 **BETWEEN 10 AND 50** → 10세부터 50세까지

2. [컬럼] >= A and [컬럼] <= B

ex) 나이 **>= 10 and 나이 <= 50** → 10세부터 50세까지

답 : ① BETWEEN ② AND

PATH 11**1. 에러(Error)**

- 소프트웨어 개발 / 유지 보수 중에 발생한 부정확한 결과
- 개발자 실수, 개발 명세서의 잘못된 이해, 서브루틴 기능 오해 등

2. 오류(Fault)

- 프로그램 코드 상에 존재하는 것
- 정상/비정상적인 프로그램 버전간의 차이로 인하여 발생

3. 실패(Failure)

- 정상/비정상적인 프로그램 버전간의 실행 결과의 차이
- 실제 실행 결과를 개발 명세서의 예상 결과와 비교함으로써 발견

4. 결함(Defect)

- 버그, 에러, 오류, 실패, 프로그램 실행에 대한 문제점, 프로그램 개선 사항 등의 전체를 포괄하는 용어이다.

답 : 결함(Defect)

PATH 12**1. MySQL**

- SQL을 사용하는 오픈 소스 관계형 데이터베이스 관리 시스템
- 오라클 사가 관리 및 지원
- 다중 사용자 및 다중 스레드 형식을 지원
- C++, java, PHP, Python 스크립트 등을 위한 응용 프로그램 인터페이스(API)를 제공
- 유닉스나 리눅스, 윈도우 운영 체제 등에서 사용 가능

답 : MySQL

PATH 13**1. 배열(Array)**

- 같은 자료형을 가진 변수들의 집합입니다.

Array [**index_number**] = {Value1, Value2...}

↳ 0부터 시작

2. 코드분석

```
for(int i=0; i<sizeof(num)/sizeof(int); i++)
// sizeof란 바이트의 크기를 구하는 함수이다. (20 / 4) = 5
    for(int j=0; j<i; j++) // j를 i번만큼 반복
        if(num[i] >= num[j]){
            temp = num[i];
            num[i] = num[j];
            num[j] = temp;
        }
// 해당 조건에 부합할 경우 temp 변수를 이용한 스왑(최대값부터)
    }
```

답 : 50 42 23 19 1

PATH 14**1. 프로젝트(PROJECT, Π)**

- 릴레이션 내에서 조건을 만족하는 **속성**을 구하기 위한 연산
- 결과문은 조건을 만족한 속성들로만 구성

2. 셀렉트(SELECT, σ)

- 릴레이션 내에서 조건을 만족하는 **튜플**을 구하기 위한 연산.
- 결과문은 조건을 만족한 튜플들로만 구성

3. 조인(JOIN, \bowtie)

- 두 테이블로부터 조건에 맞는 튜플만을 하나의 튜플로 결합
- 종류 : 동일조인, 자연조인, 외부조인, 세타조인 등

답 : ① 이름, 점수 ② 과목='수학'

PATH 15**1. 아이노드(i-node)**

- 유닉스에서 각각의 파일을 관리하기 위해 사용되는 자료 구조

2. 아이노드 유형(링크파일)**1) 심볼릭 링크(Symbolic Link) = 소프트 링크**

- 윈도우의 바로 가기(.LNK)와 비슷한 개념
- 실제 파일의 대한 경로를 가지고 있는 파일
- 구조가 다른 파일 시스템 내에서도 링크가 가능

2) 하드 링크(Hard Link)

- 원본 파일과 동일한 파일을 추가/수정/삭제(→심볼릭 링크)

답 : i-node**PATH 16****1. switch case문(조건문)**

- 변수 값을 확인하여, case 조건과 일치할 시 실행문을 인식
- case 조건은 콜론(:)으로 구분하며, break로 처리를 종료합니다.
- default는 If문의 else와 같은 역할을 합니다.

2. 코드 분석

```
switch(num){ // num = 1 1(숫자) '1'(문자)는 다릅니다.
    case '1': // num == '1' (조건 부합하지 않음)
        printf("%d", ++num + 2); break; // 3
    case '2': // num == '2' (조건 부합하지 않음)
        printf("%d", num++ + 1); break; // 2
    case '3': // num == '3' (조건 부합하지 않음)
        printf("%d", ++num + 3); break; // 4
    default: // 나머지의 값 (조건 부합)
        printf("%d", num); break; } // 1
```

답 : 1

PATH 17**1. 데이터베이스 트랜잭션(Transaction)**

- 하나의 작업을 수행하기 위해 필요한 연산들의 집합
- 데이터베이스에서 논리적인 작업의 단위
- **트랜잭션의 특성(ACID)**

원자성	일관성	독립성	지속성
Atomicity	Consistency	Isolation	Durability

답 : 지속성(영속성)

PATH 18**1. 증감 연산자(+/-)**

- 증감 연산자는 위치에 따라 전위와 후위 연산자로 구분.
- 전위 연산자($++i$ / $--i$) : 피연산자 증감 후, 해당 라인 연산 수행
- 후위 연산자($i++$ / $i--$) : 해당 라인 연산 수행 후, 피연산자 증감

2. 코드분석

```

int function(int num){ // 서브 함수 생성
    ++num; // num의 값 1 증가 ( num = 6 )
    return num++; // num 값 반환 ( ++ 인식 불가 ) }

void main(){
    int num=5;
    num = function(num); // 서브 함수 호출
    num++; // 다음 줄부터 값이 1 증가
    printf("%d", num); } // 7 출력
  
```

답 : 7

PATH 19**1. Windows 단축키**

- **바탕화면 표시 : 윈도우키 + D**

※ 재입력시, 실행 창이 원위치로 돌아옵니다.

- 창 최소화 : 윈도우키 + M

※ 최소화한 창을 다시 돌리려면 윈도우키 + Shift + M

답 : 윈도우키 + D

PATH 20**1. 아키텍처(Architecture)**

- 시스템 아키텍처와 소프트웨어 아키텍처로 구분

2. 소프트웨어 아키텍처(Software Architecture)

- 소프트웨어의 골격이 되는 기본 구조

- **구성요소(Component) 간의 관계를 표현하는 시스템 구조/구조체**

- 설계 기본원리 : 모듈화 / 추상화 / 단계적 분해 / 정보은닉

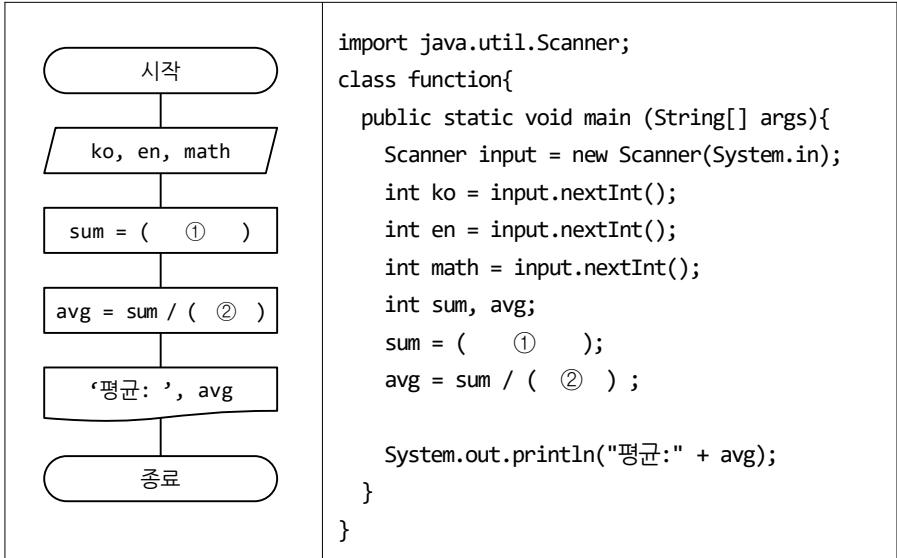
**답 : 소프트웨어 아키텍처
(Software Architecture)**

PATH

**정보처리기능사 실기
예상문제 5회**

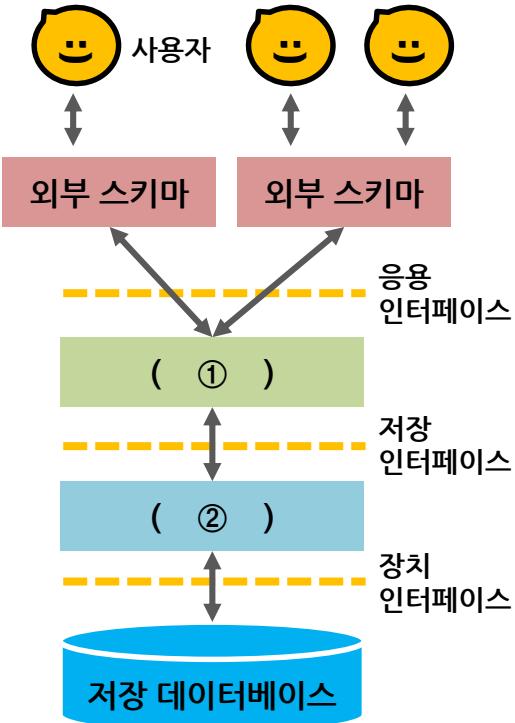
01.

국어, 수학, 영어점수의 평균을 출력하는 프로그램이 있다.
아래의 괄호 안에 적합한 표현을 작성하시오.



02.

다음의 그림을 보고 빈칸에 해당하는 용어를 쓰시오.



03. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오.

하나의 PC에 여러 개의 운영체제를 설치하여 개발하는 프로그램
을 다양한 환경에서 테스트 할 수 있도록 하고, 서로 다른 운영체
제에서 구동되는 프로그램을 실행시키도록 지원하는 것으로,
플랫폼 (①)와 리소스 (①)로 구분된다.

04. USB 포트 등의 연결장치를 이용해 컴퓨터에 하드웨어를 연
결할 시 사용자의 별도 조작이나 드라이버 설치 없이도 바로
사용할 수 있게 해주는 기능은 무엇인가?

- 05.** 아래의 <출석>테이블에서 3학년인 학생들의 이름을 중복되지 않게 검색하는 SQL문을 완성하시오.

학년	이름	과목	평균	등급
1	허원무	국어	64	C
3	정선후	영어	59	D
2	전은수	과학	82	A
2	윤종호	수학	76	B

```
SELETE ( ①  )
FROM 출석
WHERE 학년 = 3;
```

- 06.** 다음은 학생들의 성적을 나타낸 표이다.
아래의 릴레이션을 보고, 기수(Cardinality)를 구하시오.

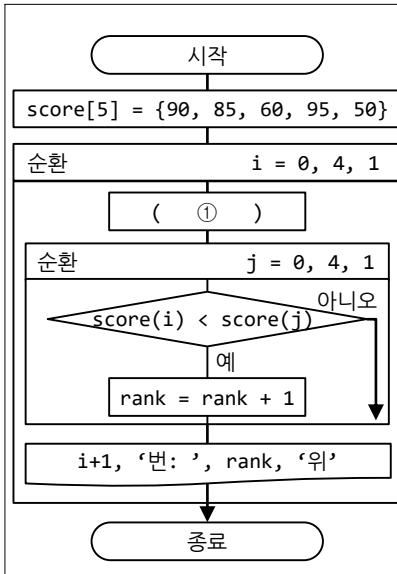
순번	이름	과목	점수	등급
1	나중빈	화학	84	2
2	정해민	물리	93	1
3	노영준	화학	82	2
4	이민우	생물	77	3

07. 다음에서 설명하는 용어를 쓰시오.

(①) 프로토콜은 네트워크 상에서 서버와 클라이언트 간의 데이터를 전송 및 교환을 하기 위해 사용하게 되며, 익명 서버를 제외한 해당 서버에 접속하기 위해서는 계정/암호를 필요로 한다.

08. Windows 내의 명령어 중 tracert와 같은 기능을 하는 UNIX의 명령어는 무엇인가?

- 09.** 배열로 입력 받은 점수의 순위를 출력하는 프로그램이 있다.
아래의 괄호 안에 적합한 표현을 작성하시오.



```
#include<stdio.h>
int main(){
    int score[5] = {90, 85, 60, 95, 50};
    int rank;

    for(int i=0; i<5; i++){
        ( ① );
        for(int j=0; j<5; j++)
            if(score[i]<score[j]){
                rank++;
            }
        printf("%d번: %d위\n", i+1, rank);
    }
}
```

- 10.** '김동원'에게 학생 테이블에 대한 검색(SELECT) 권한 회수 명령어의 예제 빈칸을 완성하시오.

```
REVOKE SELECT ON 학생 ( ① ) 김동원;
```

11.

다음에서 설명하는 용어를 쓰시오.

네트워크 클래스는 IPv4 주소의 부족으로 인하여 각 사용 목적에 따라 주소 대역대를 할당하여 Network IP와 Host IP를 구분 짓는 방법이다. 이 중 (①)클래스는 기업 안의 소규모 네트워크에 적용하며, 첫 옥텟이 192로 시작하게 되는 통신 대역대이다.

12.

다음과 같이 C 언어로 구현된 프로그램을 분석하여 그 실행결과를 작성하시오.

```
#include<stdio.h>
int function(int num){
    ++num;
    return num++;
}

int main(){
    int num1 = 5;
    int num2 = 4;
    int result;

    result = function(num1) + num2++;
    printf("%d %d %d", result, num1, num2);
}
```

13. 다음은 데이터베이스 정의어(DDL)의 종류와 그에 대한 설명이다. 빈칸에 알맞은 단어를 적으시오.

CREATE : 데이터베이스, 테이블 등 생성

ALTER : 테이블 수정

DROP : 데이터베이스, 테이블 등 삭제

(①) : 테이블 초기화

14. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int num1 = 12;
    int num2 = 8;
    int size, result;

    size = num1 > num2 ? num2 : num1;

    for(int i=1; i <= size; i++)
        if(num1%i == 0 && num2%i == 0)
            result = i;
    printf("%d", result);
}
```

- 15.** 다음의 SQL 구문에서, 삭제하려는 요소가 다른 개체를 참조하고 있을 때는 삭제되지 않게 하는 명령어를 완성하시오.

```
DROP TABLE 명단 ( ① )
```

- 16.** 다음에 제시된 프로그램은 입력 받은 값(분)을 시와 분으로 변환하여 출력한다. 빈칸 적합한 연산자를 각각 작성하시오.

```
#include<stdio.h>
int main(){
    int num, hour, minute;
    printf("Minute : ");
    scanf("%d", &num);

    if(num >= 60){
        hour = num ( ① ) 60;
        minute = num ( ② ) 60;
        printf("%d시간 %d분입니다.", hour, minute);
    }
    else{
        printf("%d분입니다.\n", num);
    }
}
```

17. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main()
{
    int num1, num2, result;
    num1 = num2 = 25;
    result = num1 + ~num2;

    printf("%d %d %d", num1, ~num2, result);
}
```

18. 다음의 설명하는 애플리케이션 테스트 방법은 무엇인가?

(①)는 프로그램의 내부 로직(수행 결로 구조, 루프 등)을 보면서 테스트를 수행하는 방법이다. (①)의 종류로는 루프테스트, 조건 테스트, 분기테스트 등이 있다.

19. 다음은 UML 중 유스케이스 다이어그램의 구성 요소에 대한 설명이다. 보기에서 설명하는 구성요소는 무엇인가?

- 사용자의 시점에서 시스템을 모델링 하는 역할
- 어떤 시스템을 만드느냐를 사용자 입장에서 우선해서 시스템이 어떠해야 하는가를 알아보며 결정하는 것

20. 다음의 설명하는 테스트 오라클의 유형으로 알맞은 것은?

테스트 오라클의 한가지 유형인 (①)은, 특정한 몇 개의 입력 값에 대해서만 기대하는 결과를 제공해 준다.
전수 테스트가 불가한 경우에 사용하며, 경계값, 구간별 예상 값 결과 작성 시 사용한다.

1.	① ko + en + math ② 3
2.	① 개념스키마 ② 내부스키마
3.	가상화
4.	PnP(Plug&Play)
5.	DISTINCT 이름
6.	4개
7.	FTP(File Transfer Protocol)
8.	TRACEROUTE
9.	rank = 1
10.	FROM

11.	C
12.	10 5 5
13.	TRUNCATE
14.	4
15.	RESTRICT
16.	① / ② %
17.	25 –26 –1
18.	화이트박스 테스트
19.	유스케이스(Use case)
20.	샘플링 오라클 (Sampling Oracle)

PATH 01**1. 입력문(Import java.util.scanner)**

- 입력을 담당하는 헤더파일(클래스)
- println() 과 달리 객체를 생성해야만 사용 가능

2. 객체 생성

- Scanner 객체명 = new Scanner(System.in)을 사용

3. 객체 사용

- (자료형) 변수 = 객체명.next자료형()

```
import java.util.Scanner; // 입력 관련 헤더파일
class function{
    public static void main (String[] args){
        Scanner input = new Scanner(System.in); // 입력 객체 선언
        /*----- 중략 -----*/
        sum = ko + en + math; // 입력 받은 값을 더함
        avg = sum / 3; // 평균 계산(합÷수)
        System.out.println("평균:" + avg);
    }
}
```

답 : ① ko + en + math ② 3

PATH 02**1. 외부스키마 = 서비스키마 = 사용자 View**

- 개인(사용자나 응용프로그래머 등)의 관점
- DB에서 각 사용자가 필요로 하는 부분만 정의
- 여러 개의 외부 스키마가 존재 가능

2. 개념스키마 = 논리스키마 = 전체적 View

- 전체 조직적인 관점(기관, 조직)
- 개체간의 관계 / 제약조건을 정의
- 하나의 개념스키마만 존재 / DBA에 의해 정의

3. 내부스키마 = 물리스키마

- 시스템프로그래머의 관점
- 물리적 저장장치의 입장에서 본 DB구조
- 개념 스키마를 기억장치에 물리적으로 구현하기 위한 방법 기술

답 : ① 개념스키마 ② 내부스키마

PATH 03**1. 플랫폼 가상화**

- 하드웨어 플랫폼 위에서 실행되는 호스트 프로그램이 게스트 프로그램을 만들어서 마치 독립된 환경인 것처럼 보여주는 것.

2. 리소스 가상화

- 메모리, 저장장치, 네트워크 등을 결합하거나 나누는 것.
- 사용자는 가상화된 물리적 장치의 위치를 모르는 경우가 대부분

답 : 가상화**PATH 04****1. Windows PnP(Plug and Play)**

- 외부장치를 연결(Plug)하면 자동으로 프로그램을 찾아 실행(Play)하기 때문에, 하드웨어 연결과 관련된 설정과 설치가 불필요
- Windows 95 버전부터 지원

답 : PnP(Plug&Play)

PATH 05

1. 일반 검색(SELECT)

```
SELECT 이름  
FROM 출석  
WHERE 학년 = 3;
```

- 결과 : 출석 테이블에서 학년이 3학년인 대상의 이름을 불러옵니다(중복된 값 발생 가능).

2. 중복제거 검색(SELECT DISTINCT)

```
SELECT DISTINCT 이름  
FROM 출석  
WHERE 학년 = 3;
```

- 결과 : 출석 테이블에서 학년이 3학년인 대상의 이름을 중복 없이 불러옵니다.

답 : DISTINCT 이름**PATH 06**

1. 속성(Attribute)

Red	White	Red	White	Red
White	Red	White	Red	White
Red	White	Red	White	Red

2. 튜플(Tuple)

Red				
Red				
Red				

릴레이션의 속성(Attribute)의 개수를 차수(Degree)라 하며, 튜플(Tuple)의 개수를 카디널리티(Cardinality:기수)라 부릅니다.

※ 본 문제의 속성은 5개이므로 차수는 5 입니다.

첫 행을 제외한 튜플의 수는 총 4줄이므로 카디널리티는 4 입니다.

답 : 4개

PATH 07

1. FTP(File Transfer Protocol)

- 파일 등의 데이터들을 전송 및 교환에 사용되는 프로토콜.
- FTP 서버에 접속하기 위해, 계정과 암호가 필요합니다.
※ Anonymous FTP는 익명성을 보장합니다.

2. TELNET

- 네트워크 기반의 원격 통신 프로토콜
- 데이터 전송 시 암호화 작업이 없어서, 보안상 위험성이 높습니다.

3. SSH(Secure Shell)

- 원격 통신에 이용되는 프로토콜
- 데이터 전송 시 암호화 기법을 통해, 보안상의 위험성이 낮습니다.

답 : FTP(File Transfer Protocol)

PATH 08

1. TRACEROUTE(경로 추적)

- 목적지까지 경유하는 시스템의 정보를 확인
- `traceroute`의 약어 : `tracert(Win)`
- 경유하는 라우터의 개수, IP와 도착시간 등을 수집

답 : TRACEROUTE

PATH 09

1. 코드분석

```
#include<stdio.h>
int main(){
    int score[5] = {90, 85, 60, 95, 50};
    int rank;

    for(int i=0; i<5; i++){    //기준값 설정(배열은 0번부터)
        rank = 1;    //순위 초기화
        for(int j=0; j<5; j++) //비교값 설정
            if(score[i]<score[j]){ //기준값 < 비교값
                rank++;    //순위 증가
            }
        printf("%d번: %d위\n", i+1, rank); //ex> 1번: 2등
    }
}
```

답 : rank = 1

PATH 10

1. 권한 해제(REVOKE)

```
REVOKE SELECT ON 학생 FROM 김동원
```

- 결과 : 김동원에게 학생 테이블 검색 권한을 해제합니다.

2. 권한 부여(GRANT)

```
GRANT SELECT ON 학생 TO 김동원
```

- 결과 : 김동원에게 학생 테이블 검색 권한을 부여합니다.

답 : FROM

PATH 11**1. 네트워크 클래스(Classful network)**

- 네트워크 단말의 증가로 가용 가능한 IPv4의 주소가 부족해졌고,
이에 **사용 목적에 따라 IP의 대역대를 나누어** 각 규모에 따라
관리하기 쉽게 표현한 것을 “클래스”라 합니다.

Class	IP주소의 첫째 옥텟	사용 목적	이론적 IP 주소 범위
A class	0xxx xxxx	대륙 안통신	0.0.0~ 127.255.255.255
B class	10xx xxxx	국가안통신	128.0.0 ~ 191.255.255.255
C class	110x xxxx	기업 안통신	192.0.0 ~ 223.255.255.255
D class	1110 xxxx	그룹통신	224.0.0 ~ 239.255.255.255
E class	1111 xxxx	연구용통신	240.0.0 ~ 255.255.255.255

답 : C**PATH 12****1. 증감 연산자**

- 피연산자를 1씩 증가(혹은 감소)시켜 주는 구문을 의미합니다.
- 위치에 따라 전위와 후위로 나누게 됩니다.
ex) ++피연산자 : 1의 값 증가 이후, 해당 라인 연산
 피연산자-- : 해당 라인 연산 이후, 1의 값 감소

2. 코드 분석

```

int function(int num){ // 호출할 함수 생성
    ++num; // 6
    return num++; // 값 6을 반환 이후에 증감 연산, 값 변화X
}
int main(){
    int num1 = 5;
    int num2 = 4;
    int result;
    result = function(num1) + num2++; // 6 + 4
    printf("%d %d %d", result, num1, num2); // 10, 5, 5
} // num1의 경우 function(num1)을 변수에 저장하지 않았기에
  // 값이 변하지 않는다.
  
```

답 : 10 5 5

PATH 13

1. TRUNCATE

TRUNCATE TABLE [테이블명]

- 테이블 내 데이터를 전부 삭제하며,
사용하고 있던 공간을 반납하는 명령어
- 테이블의 데이터만 모두 삭제될 뿐, **테이블 자체는 보존**
(DROP은 테이블 자체를 삭제)

답 : TRUNCATE

PATH 14

1. 삼항 연산자

- 조건(논리연산)에 부합할 경우, True와 False에 해당하는 값을 출력하는 구문을 의미합니다.

- 형식 = 조건 ? True : False

ex) `10 > 3 ? 'A' : 'B'` ▶ 'A' 출력

2. 증감 연산자

- 피연산자를 1씩 증가(혹은 감소)시켜 주는 구문을 의미합니다.
 - 위치에 따라 전위와 후위로 나뉘게 된다.
- ex) ++피연산자 : 1의 값 증가 이후, 해당 라인 연산
피연산자-- : 해당 라인 연산 이후, 1의 값 감소

3. 코드 분석

```
size = num1 > num2 ? num2 : num1; // 12 > 8 ? 8 : 12
// 두 수를 비교하여 작은 수를 size에 대입하여 최소로 반복
for(int i=1; i <= size; i++) // 1부터 8까지 반복
    if(num1 % i == 0 && num2 % i == 0) { // 조건 모두 만족시 인식
        result = i;
    }
printf("%d", result); // 4 출력
```

답 : 4

PATH 15

1. RESTRICT : 제한, 한정

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고
있을 경우 **변경 또는 삭제가 취소됩니다.**

2. CASCADE : 종속

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고
있을 경우 연쇄적으로 함께 변경 또는 삭제합니다.

답 : RESTRICT

PATH 16

1. 산술 연산자

- 특정 기호를 이용하여 사칙연산을 하는 작업을 의미합니다.
- **+ [덧셈], - [뺄셈], * [곱셈], / [나눗셈], % [나머지]** 등이 있습니다.

2. 코드 분석

```
if(num >= 60){ // num값이 60보다 커울 경우 인식
    hour = num / 60; // hour에 num을 60으로 나눈 값 대입
    // ex) 325 / 60 = 5
    minute = num % 60; // minute에 나머지를 대입
    // ex) 325 / 60 = 25
    printf("%d시간 %d분입니다.", hour, minute); }

else{ // 위의 if 조건이 부합하지 않으면 인식
    printf("%d분입니다.\n", num); }
```

답 : ① / ② %

PATH 17**1. 보수 연산자**

- 이진수의 음수 표현을 위해 사용
- C언어 보수 연산자 경우, 값의 앞에 '~' 표시

a. 1의 보수 : 0을 1로, 1을 0으로 변경

ex) 0000 1100(12) → 1111 0011(-12)
// 8비트의 가장 처음 값은 부호의 역할

b. 2의 보수 : 1의 보수에 +1

ex) 1111 0011(-12) → 1111 0011 + 0000 0001 = -(12+1)

2. 코드 분석

```
int num1, num2, result;  
num1 = num2 = 25; // num1과 num2의 25 대입  
result = num1 + ~num2; // 25 + -26  
  
printf("%d %d %d", num1, ~num2, result);
```

답 : 25 -26 -1

PATH 18**1. 화이트박스 테스트**

- **개발자 관점**으로 **내부**구조와 동작을 테스트합니다.

- 종류

- | | | |
|-----------|------------|---------|
| · 기초경로테스트 | · 제어흐름테스트 | · 조건테스트 |
| · 루프테스트 | · 데이터흐름테스트 | · 분기테스트 |

2. 블랙박스 테스트

- **사용자 관점**으로 명세 기반(외부)의 테스트 방법입니다.

- 종류

- | | |
|--------------|----------|
| · 균등분할 | · 한계값테스트 |
| · 원인효과그래프테스트 | · 비교테스트 |

답 : 화이트박스 테스트

PATH 19**1. 통합 모델링 언어(Unified Modeling Language)**

- 개발자간의 의사소통을 원활히 하기 위한 표준화한 모델링 언어

2. 유스케이스 다이어그램 (Use Case Diagram)

- 사용자-시스템 간 상호작용의 표현
- 시스템범위, 엑터, **유스케이스**, 관계로 구성
System Scope Actor Usecase Relationship

3. 유스케이스(Use case)

- 시스템의 동작을 **사용자의 입장**에서 표현한 시나리오
- 시스템에 관련한 요구사항을 알아내는 과정

답 : 유스케이스(Use case)

PATH 20**1. 테스트 오라클**

- 테스트의 결과가 참인지 거짓인지를 판단하기 위해서 **사전에 정의된 참 값을 대입하여 비교**하는 소프트웨어 테스트 기법 및 활동

2. 유형**a. 참 오라클(True Oracle)**

- 모든 입력 값의 기대 결과를 생성해서 발생된 오류를 모두 검출
- 전수 테스트 가능

b. 샘플링 오라클(Sampling Oracle)

- 특정한 입력 값들에 대해서만 기대하는 결과를 제공
- 전수테스트가 불가할 경우 사용

c. 휴리스틱 오라클(Heuristic Oracle)

- 샘플링 오라클에 휴리스틱(추정값) 입력값을 추가
- 확률/직관에 의한 예상결과

d. 일관성 검사 오라클(Consistent Oracle)

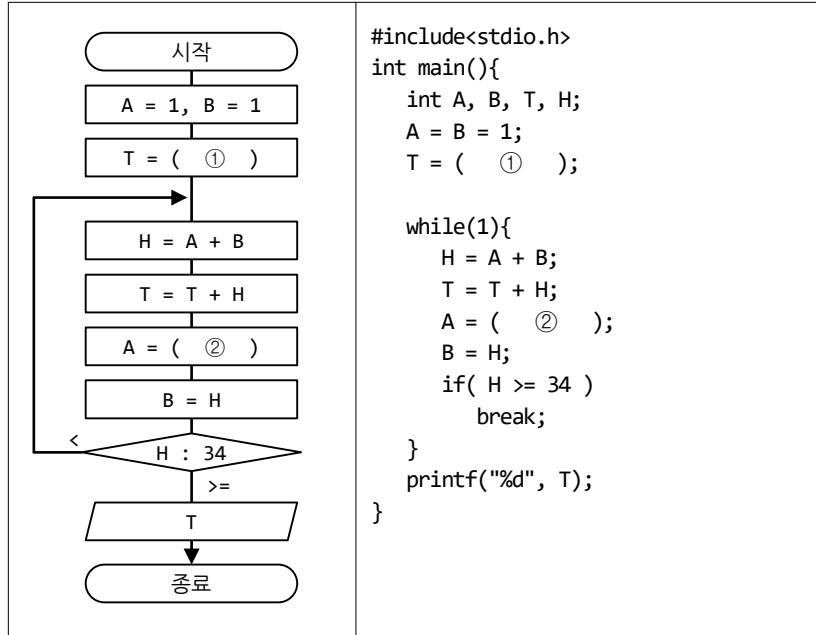
- 애플리케이션 변경 시, 수행 전과 후의 값의 동일 여부 검증

답 : 샘플링 오라클(Sampling Oracle)

PATH

**정보처리기능사 실기
예상문제 6회**

- 01.** 다음은 피보나치 수열을 누적 출력하는 프로그램이다.
아래의 괄호 안에 적합한 표현을 작성하시오.



- 02.** 다음은 성적 테이블에서 점수가 70점 이하인 학생들의 이름과 등급을 추출하는 연산에 대한 결과이다. 다음을 보고 쿼리의 빈칸에 들어갈 단어를 작성하시오.

이름	과목	점수	등급
강규안	국어	56	6
김준섭	사회	87	2
이경민	수학	63	4
박정원	국어	95	1



SELECT (①) FROM 성적 WHERE (②)

03. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오.

(①)은 커널과 프로그램 사이에서 상호작용을 가능하게 해주는 처리 프로그램을 의미하며, 사용자로부터 명령을 받아 커널이 이해하도록 해석하고 그에 대한 처리 결과를 실행시키는 역할을 수행한다.

04. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오

시스템의 동작상태를 유지한 채 기존의 장치를 제거하고 새로운 장치를 연결하면 바로 인식되어 사용할 수 있게 하는 기능으로서, 대표적으로 항상 전원을 유지해야 하는 서버 컴퓨터나 RAID 시스템 등이 있다.

05.

다음에서 설명하는 용어를 쓰시오.

- 사용자가 시스템에 제기한 서비스에 대한 작업 수행을 위한 데이터 조작어의 집단
- 데이터베이스의 일관적 상태 유지를 위한 수행 제어 및 회복 작업 시 처리되는 단위
- 데이터 베이스의 상태를 참조 및 갱신 등의 작업

06.

조직 전체를 총괄적 입장에서 데이터베이스(DB)를 정의한 것으로 조직의 효율적 관리를 위하여 접근 권한, 보안정책, 무결성 규칙 등을 추가적으로 삽입한 데이터 베이스의 구조를 의미하는 것은?

07. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
main(){
    int num = 10; int i;

    for(i=1; i<=5; i++){
        if(num%i == 0)
            num++;
        else if(num%i != 0)
            num--;
    }

    printf("%d", num);
}
```

08. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오.

가상머신을 생성하고 실행할 수 있는 (①)는 단일 호스트 컴퓨터의 자원을 이용하여 다수의 운영체제를 가상으로 실행하기 위한 논리적 플랫폼을 의미한다.
대표적인 (①)로 VMWARE, Virtual Box 등이 있다.

09. 배열로 입력 받은 점수의 순위를 출력하는 프로그램이 있다.
아래의 괄호 안에 적합한 표현을 작성하시오.

```
#include<stdio.h>
int main()
{
    int num = 5;
    int sum = 0;
    for(int i=0; ; i++){
        sum += num;
        if(++num > 10)
            break;
    }
    printf("%d", sum);
}
```

10. 다음에서 설명하는 용어를 쓰시오 .

- 데이터베이스(DB)를 구성하는 가장 작은 논리적 단위이며, 더 이상 분리되지 않는다.
- 파일 구조상 데이터 항목 또는 필드에 해당한다.

11.

다음에서 설명하는 용어를 쓰시오.

물리적 배치와 상관없이 스위치에서 나누어진 가상의 논리적 네트워크를 의미하며, 여러 네트워크를 하나로 구성 가능하며, 관리하는데 유연하지만, 나누어진 정보를 서로에게 전달 할 수가 없기에 라우터를 이용하여 이를 보완한다.

12.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int function(int *arr);
int main(){
    int num[5] = {2,5,7,9,10};
    function(num);
}
int function(int *arr){
    int comp, comp2;
    for(int i=1; i<5; i++){
        comp = arr[i] - arr[i-1];
        if(i == 1)
            comp2 = comp;
        if(comp < comp2)
            comp2 = comp; }
    for(int j=1; j<5; j++)
        if(comp2 == arr[j] - arr[j-1])
            printf("%d, %d", arr[j-1], arr[j]);
}
```

13.

다음에서 설명하는 용어를 쓰시오 .

- 하나 이상의 원하는 데이터만을 제한적으로 보기 위해 가상의 테이블이다.
- 저장되어 있지는 않지만 사용자에게 있는 것처럼 간주한다.
- 기본 테이블과 거의 동일하게 사용할 수 있으나,
- 삽입, 수정, 삭제에 제약을 받게 된다.

14.

가중치의 심각도를 측정하기 위해, 전체 시스템에 결함이 미치는 영향을 정의하는 것을 결함심각도 라고 한다.
다음 빈칸에 알맞은 결함 심각도를 작성 하시오.

우선순위	설명
(①)	단위 업무 프로세스가 비정상적으로 수행되고, 해당 프로세스로 인하여 다른 단위 기능 또는 시스템의 시스템 다운, 장시간 응답 지연, 데이터 왜곡 등 전반적인 기능에 장애가 발생하는 경우의 결함
(②)	단위 업무 프로세스가 일반 사용자 환경에서는 정상적으로 완료될 수 없는 경우의 결함
(③)	단위 업무 프로세스가 일반 사용자 환경에서는 정상적으로 완료될 수 있으나, 특정 사용자는 정상적으로 완료 될 수 없는 경우의 결함

15.

다음에서 설명하는 용어를 쓰시오 .

- 관계 데이터 베이스 관리에서 관계를 조작하기 위한 방법
- 비절차적 언어로서 무엇인지만 명시하고 질의를 어떻게 수행 할 것인가는 명시하지 않고 사용한다.
- 종류로는 튜플과 도메인 등에 사용한다.

16.

다음과 같이 C 언어로 구현된 프로그램을 분석하여 그 실행결과를 작성하시오.

```
#include<stdio.h>
int function(int arr[]){
    int max, min, sum;
    max = min = arr[0];
    sum = 0;
    for(int i=0; i<3; i++){
        if(max < arr[i])
            max = arr[i];
        if(min > arr[i])
            min = arr[i];
        sum += arr[i]; }
    sum = sum - max - min;
    return sum; }
int main(){
    int num[] = {82,61,74};
    int result;
    result = function(num);
    printf("%d", result); }
```

17.

다음에서 설명하는 용어를 쓰시오.

(①) 이란 HTTP, HTTPS SMTP 등을 통해 규격화 및 표준화가 되어 있기에 모든 기종에서 호환성을 제공하게 되며, 이를 통해 XML을 이용하여 이기종 간의 정보교환에 사용되는 프로토콜이다.

18.

다음의 설명하는 용어는 무엇인가?

작성한 개발소스 코드를 분석하여 개발 표준에 위배되었거나 잘못 작성된 부분을 수정하는 작업을 말한다.

이 작업을 통해 소프트웨어의 품질을 높일 수 있지만 시간이 다소 많이 걸린다는 단점이 있다.

19. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int num = 7;
    int bin[8] = {0, };

    for(int i=0; ; i++){
        bin[i] = num % 2;
        num = num / 2;

        if(num == 0)
            break;
    }
    for(int j= 7; j>=0; j--)
        printf("%d", bin[j]);
}
```

20. 다음 설명하는 테스트 기법은?

다양한 코드중심 테스트 프레임 워크는 각각의 테스트 프레임워크를 가지고 있다. 그 중 이것은 자바 프로그래밍 언어용 유닛 단위 테스트 프레임워크로 단정문을 이용하여 테스트 케이스의 수행결과를 판별한다.

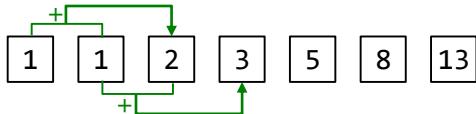
1.	① A + B ② B
2.	① 이름, 등급 ② 점수 <= 70
3.	쉘(Shell)
4.	핫스왑(Hot Swap)
5.	트랜잭션(Transaction)
6.	개념 스키마
7.	7
8.	하이퍼바이저(Hypervisor)
9.	45
10.	속성(Attribute)

11.	VLAN(Virtual LAN)
12.	9, 10
13.	뷰(View)
14.	High/Medium/Low
15.	관계 해석
16.	74
17.	SOAP (Simple Object Access Protocol)
18.	코드 인스펙션
19.	00000111
20.	Junit

PATH 01**1. 피보나치 수열**

- 앞의 **두 수의 합이 뒤의 있는 수**가 되는 계산 방식을 의미합니다.

ex)

**2. 코드 분석**

```

int A, B, T, H;
A = B = 1; // A 와 B에 값 1 저장
T = A + B; // T: 누적 값 저장변수
      (반복문 실행시, 첫 두개 수의 합계가 누적되지 않기 때문에)
while(1){ // 무한 루프(반복)
    H = A + B; // 뒤의 있는 수를 위한 변수
    T = T + H; // 값 누적.
    A = B;
    B = H; // 뒤의 있는 수 B에 저장
    if ( H >= 34 ) // 뒤의 있는 수가 34이상일 시 정지
        break;
    printf("%d", T); } // 누적값 출력
  
```

답 : ① A + B ② B

PATH 02**1. 일반 검색(SELECT [field name])**

```

SELECT 이름
FROM 성적
WHERE 점수 <= 70;
  
```

- 결과 : 성적 테이블에서 점수가 70 이하인 대상의 이름을 불러옵니다(결과값의 열이 하나).

2. 복수 열 검색(SELECT [field name1], [field name2]...)

```

SELECT 이름, 등급
FROM 성적
WHERE 점수 <= 70;
  
```

- 결과 : 성적 테이블에서 점수가 70 이하인 대상의 이름과 등급을 불러옵니다(결과값의 열이 둘).

답 : ① 이름, 등급 ② 점수 <= 70

PATH 03

1. **쉘(Shell)**

- 커널과 사용자 간의 대화를 가능하게 하는 명령어 해석기

2. **커널(Kernel)**

- 운영체제의 핵심(중심)

- 소프트웨어와 하드웨어간의 상호 작용을 관리하는 프로그램

ex) 게임 실행 시, 게임(소프트웨어)에서 하드웨어의 성능을 조절

답 : 쉘(Shell)

PATH 04

1. **핫스왑(Hot Swap)**

- 시스템 전원이 켜진 상태에서 안전하게 장치를 제거가 가능하며, 새로운 장치를 연결할 수 있는 기능입니다.

ex) 핫스왑 베이 등을 이용하여 HDD(하드디스크), SSD 등 교체

2. **핫플러그(Hot Plug)**

- 시스템 전원이 꺼진 상태에서 제거 및 새로운 장치를 연결해야하며, 만약 켜진 상태에서 장치를 제거하면 데이터 손실 우려가 있습니다.

ex) USB

답 : 핫스왑(Hot Swap)

PATH 05**1. 데이터베이스 트랜잭션(Transaction)**

- 하나의 작업을 수행하기 위해 필요한 연산들의 집합
- 데이터베이스에서 논리적인 작업의 단위

2. 트랜잭션의 특성(ACID)

원자성(Atomicity) : 실행 중 중단되지 않게 하는 능력

일관성(Consistency) : 실행의 완료 후 데이터베이스 상태를 유지

격리(Isolation) : 한 트랜잭션의 실행 중 다른 트랜잭션이 끼어들지 못하게 보장

지속성(Durability) : 실행에 성공한 트랜잭션은 영원히 반영

답 : 트랜잭션(Transaction)

PATH 06**1. 외부스키마 = 서비스키마 = 사용자 View**

- 개인(사용자나 응용프로그래머 등)의 관점
- DB에서 각 사용자가 필요로 하는 부분만 정의
- 여러 개의 외부 스키마가 존재가 가능

2. 개념스키마 = 논리스키마 = 전체적 View

- 전체 조직적인 관점(기관, 조직)
- 개체간의 관계 / 제약조건을 정의
- 하나의 개념스키마만 존재 / DBA에 의해 정의

3. 내부스키마 = 물리스키마

- 시스템프로그래머의 관점
- 물리적 저장장치의 입장에서 본 DB구조
- 개념 스키마를 기억장치에 물리적으로 구현하기 위한 방법 기술

답 : 개념 스키마

PATH 07**1. 증감연산자(값을 증가, 감소 시키는 연산자)**

- 전위연산자(++a / --a) : 피연산자 증감 후, 해당 라인 연산 수행

```
a = 5
printf("%d", ++a) => 6, --a => 4
```

- 후위연산자(a++ / a--) : 해당 라인 연산 수행 후, 피연산자 증감

```
a = 5
printf("%d", a++) = 5, --a => 5
```

2. 코드 분석

```
int num = 10; // num에 10 대입
int i; // 반복문의 변수 선언
for(i=1; i<=5; i++){ // 5번 반복
    if(num%i == 0) // 10 나머지 i가 0일 때
        num++; // 값을 1 증가 (후위)
    else if(num%i != 0) // 10 나머지 i가 0이 아닐때
        num--; } // 값을 1 감소 (후위)
```

답 : 7**PATH 08****1. 하이퍼바이저(Hypervisor)**

- 호스트 컴퓨터의 하드웨어 등의 자원 등을 이용하여, 현재 사용중인 운영체제 이외의 다수의 운영체제를 동시에 실행 할 수 있도록 해주는 가상 플랫폼 기술.
- 대표적으로 Oracle 사의 Virtual BOX와 VMWare 등이 있습니다.

답 : 하이퍼바이저(Hypervisor)

PATH 09

1. 반복문(for문)

- 특정한 구문을 여러 번 반복하기 위해 사용되는 구문을 의미합니다.
- 일반적인 for문은 조건 값을 입력하여 반복을 종료합니다.
하지만, 조건값을 비워 놓게 되면 제한이 없기에 무한 반복이 이루어 지며, 이를 무한 루프라 합니다.

ex> `for(int i = 0; ;i++)`

TIP) while문 또한, 조건란을 while(1)와 작성시 무한 루프가 실행됩니다.)

2. 코드 분석

```
int num = 5; // num 5 대입
int sum = 0; // sum 0 대입
for(int i=0; ; i++){ // i는 0부터 무한 반복
    sum += num; // 누적값 sum
    if(++num > 10) // 증가된 num이 10을 초과시 정지(break)
        break;
}
printf("%d", sum); // 5~10까지의 합계 출력
```

답 : 45

PATH 10

1. 속성

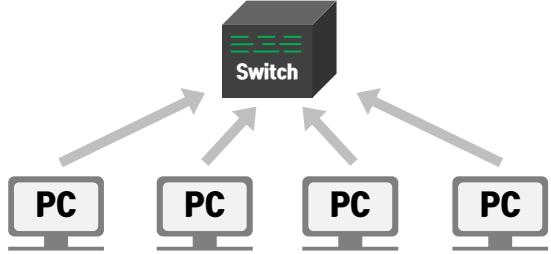
학과	학년	성명
산업디자인	2	정수한
컴퓨터공학	1	이정수

— 속성
} 객체
인스턴스

- 데이터 표현의 최소의 단위이며,
하나의 인스턴스에서 단 하나의 값만을 가질 수 있습니다..
- 파일의 구조상 데이터 항목, 데이터 필드 등에 해당하게 됩니다..
종류로 기본 속성, 설계 속성, 파생 속성 등이 있습니다.

답 : 속성(Attribute)

PATH 11



1. 가상 LAN(Virtual LAN)

- 스위치 장비에서 제공되는 논리적 네트워크를 구성하는 기능
- 하나의 브리지(Bridge)를 통해 복수의 논리적 네트워크를 구성

2. 장단점

- LAN을 세분화시켜 트래픽 감소 기능 / 네트워크 구성에 유연
- 복수 네트워크가 하나의 장비에 존재하므로,
하나의 물리적 장애가 발생 시에 다른 네트워크까지 영향

답 : VLAN(Virtual LAN)

PATH 12

1. 포인터 변수

- 변수의 메모리 주소 등을 저장 및 참조하는 변수
(*변수 : 포인터 변수 선언, & : 주소값 출력)

포인터 변수

int *ptr = &value1

0x7ffc02b180b4

변수명: value1

value2

값:

9

10

주소: 0x7ffc02b180b4 0x7ffc02b180b8

printf("%p", ptr) = 0x7ffc02b180b4 (value1의 주소)
printf("%d", *ptr) = 9 (value1의 값)

2. 코드 분석

```

int comp, comp2; // 비교값을 저장할 변수 선언
for(int i=1; i<5; i++){ // 1~4번 반복
    comp = arr[i] - arr[i-1];
    // i-1이 i보다 작기에 부호없는 정수 출력을 위해 선언
    if(i == 1) // 가장 처음 반복 시에 값 저장 용도
        comp2 = comp;
    if(comp < comp2) // 차이값이 가장 적은 값 저장
        comp2 = comp; }
  
```

답 : 9, 10

PATH 13**1. 뷰(View)**

- 사용자가 허용한 데이터만을 제한적으로 보기 위해,
하나 이상의 자료를 통해 **가상의 테이블**을 만드는 것.
- 일반 테이블과 유사하게 사용 가능하지만,
ALTER문 등을 사용할 수 없어서 데이터의 조작에 제약 존재.
(기본키를 포함한 속성을 집합으로 구성하면 가능합니다.)
- 뷰의 경우는 변경이 불가능하며, 삭제 후에 재생성 해야 합니다.

답 : 뷰(View)**PATH 14****1. 결함 심각도(Defect Severity)**

- 애플리케이션에 발생한 결함이 전체 시스템에 미치는 영향의 척도
- High - Medium - Low로 구분

High	<ul style="list-style-type: none">▪ 프로세스를 더 이상 진행할 수 없도록 만드는 결함▪ 핵심 요구사항 미구현, 장시간 시스템 응답 지연, 시스템 다운 등
Medium	<ul style="list-style-type: none">▪ 시스템 흐름에 영향을 미치는 결함▪ 부정확한 기능, 부정확한 업무 프로세스, 데이터 베이스 에러, 보안 관련 오류 등
Low	<ul style="list-style-type: none">▪ 시스템 흐름에는 영향을 미치지 않는 결함이지만, 상황에 맞지 않는 결과나 화면구성에 대한 결함▪ 부정확한 GUI, 에러 메시지 미출력, 화면상 표시 오류

답 : High/Medium/Low

PATH 15**1. 관계 해석**

- 관계 데이터의 연산을 표현하는 방법 중의 하나로, 원하는 정보가 무엇인지만을 명시하는 **비절차적 언어**.
- 종류 : 튜플 관계 해석, 도메인 관계 해석 등등

2. 관계 대수

- 원하는 정보를 검색하기 위해서 정보를 어떻게 유도하는가를 연산자를 이용하여 정보를 구하는 **절차적 언어**.
- 종류 : SELECT, JOIN 등등

답 : 관계 해석**PATH 16****1. 코드 분석**

```
int function(int arr[]){ // 함수 선언
    int max, min, sum;
    max = min = arr[0]; // max = 82, min = 82
    // 최대값과 최소값 비교를 위한 값 저장
    sum = 0; // 누적값을 저장하기 위해 0을 선언
    for(int i=0; i<3; i++){ // 0 ~ 2번 반복
        if(max < arr[i])
            // 82보다 큰 값이 있을 시에 max에 저장
            max = arr[i];
        if(min > arr[i])
            // 82보다 작은 값이 있을 시에 min에 저장
            min = arr[i];
        sum += arr[i]; } // 82 + 61 + 74
    sum = sum - max - min;
    // sum = 271 - 82 - 61 => 74 저장
    return sum; // 74의 값을 반환
}
```

답 : 74

PATH 17**1. SOAP(Simple Object Access Protocol)**

- 네트워크 상에서 **이기종(Heterogeneous)**과 **정보통신**을 하도록 XML을 기반으로 하여 메시지를 교환하는 프로토콜입니다.

1-1. XML(eXtensible Markup Language)

- 웹 프로그래밍 언어
- 웹 상에서 글자, 문서 등의 전송을 가능하게 하는 언어
- 거의 대부분의 언어가 XML을 지원하여 호환성이 높습니다.

**답 : SOAP
(Simple Object Access Protocol)**

PATH 18**1. 코드 인스펙션(Code Inspection)**

- 개발소스 코드를 분석해 표준에 위배되거나, 규칙에 맞지 않게 구현된 부분을 수정하는 작업입니다.
- **자동화된 도구를 사용**하여 개발 및 상용 소프트웨어의 **결함을 발견 및 수정**하게 되며, 해당 프로그램의 품질을 향상시키기 위해 사용됩니다.

답 : 코드 인스펙션

PATH 19

1. 코드 분석

```
int num = 7; // num 7 대입
int bin[8] = {0, }; // 배열 생성(모든 값 0으로 초기화)

for(int i=0; ; i++){ // 무한루프
    bin[i] = num % 2; // bin[0 ~ ] = 7 %(나머지) 2
    ex) 7 % 2 = 1 => bin[0] = 1
    num = num / 2; // 7 = 7 / 2 I
                    nt(3.5) num 3 대입

    if(num == 0) // num가 0이 되었을 시에, 무한루프 탈출
        break;
}
for(int j= 7; j>=0; j--)
    printf("%d", bin[j]); // bin[7 ~ 0] 출력
```

답 : 00000111**PATH 20**

1. xUnit

- 다양한 코드중심의 테스트 프레임워크
- 소프트웨어의 함수, 클래스 등 서로 다른 구성 단위를 검사
- Junit(JAVA), CppUnit(C++), Nunit(.NET)

2. 단정문(assert~~~ 문)

- 코드표의 구문을 단정지어서('~ 이어야 한다') 처리 하는 방식

답 : Junit

PATH

**정보처리기능사 실기
예상문제 7회**

01.

다음은 C 언어로 구현된 1부터 10까지의 짝수와 홀수를 판별하는 프로그램이다. 빈칸에 알맞은 내용을 입력하시오.

```
#include<stdio.h>
int main(void)
{
    int a;

    for(int i=1; i<=10; i++){
        if ( ① )
            printf("Even %d\n", i);
        else
            printf("Odd %d\n", i);
    }
    return 0;
}
```

02.

다음은 DB 트랜잭션의 특성인 ACID에 대한 설명이다.
빈칸에 들어갈 알맞은 단어를 작성하시오.

(①)(Atomicity) : 실행 중 중단되지 않게 하는 능력
일관성(Consistency) : 실행의 완료 후 데이터베이스 상태를 유지
격리(Isolation) : 한 트랜잭션의 실행 중 다른 트랜잭션이
끼어들지 못하게 보장
지속성(Durability) : 실행에 성공한 트랜잭션은 영원히 반영

03. 다음 설명 하는 용어를 작성하시오.

소프트웨어 검토를 위해 미리 준비된 자료를 바탕해, 정해진 절차에 따라 개발자가 모집한 전문가들이 소프트웨어를 검토하는 행위로서 프로그램 실행 여부를 확인하는 정적 테스트 방법 중 하나이다.

04. 다음에서 설명하는 Windows 단축키를 쓰시오.

Windows 10의 파일 탐색기 단축키로, 라이브러리 및 문서 혹은 내 컴퓨터 등의 탐색기를 열 수 있다.

05. 다음 C 언어로 구현된 프로그램을 분석하여,
그 실행 결과를 쓰시오.

```
#include<stdio.h>

int main(){
    int x=7; int y=7; int result;
    result = x > y ? ++x : y--;
    printf("%2d%2d%2d",x, y, result);
    return 0;
}
```

06. OSI 7계층 중 하나인 네트워크 계층을 TCP/IP 4계층으로
분류했을 때, 어느 계층에 속하는가?

07. 다음의 빈칸에 들어갈 말을 작성하시오.

(①)은 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)으로, 오라클 사에서 인수해 현재 관리 및 지원을 맡고 있다.

(①)은 다중 사용자 및 다중 스레드를 지원하며 유닉스 및 리눅스, 윈도우 등의 운영체제에서 사용할 수 있다.

08. 다음의 <보기>를 참조하여, 해당 내용에 알맞은 SQL 연산자를 작성하시오.

첫번째 쿼리에서 두번째 쿼리와 공통적으로 존재하는 집합의 결과를 제외한 나머지의 집합을 보여준다.

09. 다음의 <보기를> 참조하여, 해당 내용에 알맞은 기능을 작성하시오

(①)는 준비 상태의 프로세스 중 가장 앞에 있던 프로세스가 CPU를 점유하게 되는 것을 의미하는 말로, 준비 상태에서 실행 상태로 변화하는 것을 지칭한다.

10. 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class MethodEx01 {  
    public static void main(String[] args) {  
        int num = 0;  
        sub(1, 4);  
        System.out.print(num);  
    }  
    public static int sub(int num1, int num2){  
  
        int imsub;  
        if(num2 > num1)  
            imsub = num2 - num1;  
        else  
            imsub = num1 - num2;  
        return imsub;  
    }  
}
```

- 11.** 이백수에게 회원명단 테이블에 대해 수정(UPDATE) 할 수 있는 권한을 해제하는 명령을 완성하시오.

```
REVOKE UPDATE ON 회원명단 ( ① ) 이백수
```

- 12.** 회원의 뷰를 제거하고, 연관되어 있는 뷰들을 연쇄적으로 추가 제거하고 싶을 경우, 빈칸에 옮은 단어를 작성하시오.

```
DROP VIEW 회원 ( ① )
```

13. 다음은 프로토콜(Protocol)의 기본요소에 대한 설명이다.
빈칸에 들어갈 알맞은 말을 작성하시오.

(①)(Syntax) : 데이터 형식과 부호화, 신호레벨의 규정

의미(Semantics) : 전송 조작 및 오류 제어의 정보 규정

시간(Timing) : 신호의 순서, 속도 조정 등의 규정

14. 다음은 어떠한 프로토콜에 대한 설명이다.
보기를 읽고 답을 적으시오.

- 주소 재해석 및 주소 결정 프로토콜
- IP를 이용해 MAC을 검색
- 네트워크 계층에 속해서, 데이터 링크의 MAC과 연결

15. 다음이 설명하는 단어를 적으시오.

(①)은 애플리케이션 테스트의 기본 원리 중 하나로 동일한 테스트 입력 값, 조건, 기대 결과 등으로 동일한 테스트를 반복하면 더 이상 결함이 발견되지 않는 현상을 의미한다.

16. 다음에서 설명하는 용어를 쓰시오.

- 키(Key)란 데이터베이스에서 조건에 만족하는 튜플을 찾거나 순서대로 정렬할 때 기준이 되는 속성을 말한다.
- 후보키 중에서 특별히 선정된 키로 중복된 값을 가질 수 없다.
- 유일성과 최소성을 가지며, 튜플을 식별하기 위해 반드시 필요한 키이다.

17.

다음에서 설명하는 애플리케이션 용어를 쓰시오.

(①)은 상위 모듈은 있지만 하위 모듈이 없을 때, 하위 모듈로 테스트를 진행한다. 하향식 통합 테스트에서 모듈간의 통합 테스트를 위해 일시적으로 필요한 조건만을 가지고 있으며, 실제로 동작하는 것처럼 만들어 놓은 객체이다.

18.

다음은 C언어로 구현된 프로그램으로
빈칸에 알맞은 결과 값을 입력하시오.(5점)

```
#include<stdio.h>
int main(void)
{
    int a, b, c;
    int result;
    a = 12; b = 5; c = 6;
    result = a & b | c;

    printf("%d", result);

    return 0;
}
```

19.

특정 소프트웨어 테스트 목표를 달성하기 위해 실행될 단일 테스트를 정의하는 테스트 항목에 대한 명세서로서, 경로를 실행하거나 검증하는 것을 뜻하는 단어는?

20.

MS-DOS에서 파일에 대한 속성을 변경하는 명령어로, UNIX 명령어 chmod와 동일한 기능을 하는 것은?

1.	i % 2 == 0
2.	원자성
3.	워크스루/검토회의
4.	윈도우키 + E
5.	7 6 7
6.	인터넷 계층
7.	MySQL
8.	MINUS
9.	디스패치(Dispatch)
10.	0

11.	FROM
12.	CASCADE
13.	구문
14.	ARP
15.	살충제 패러독스
16.	기본키(Primary Key)
17.	스텝(테스트 스텝)
18.	6
19.	테스트 케이스
20.	attrib

PATH 01**1. 짹수/홀수 구분법**

- 수를 **2로 나눈 나머지 1의 유무**로 짹수, 홀수를 구분할 수 있습니다.
- [C언어]는 %를 사용합니다.

2. 비교연산자(부등호)

- > 초과, \geq 이상, < 미만, \leq 이하, $!=$ 같지않다. $==$ 같다

* = 는 '값 입력' 시 사용

3. 요약

인수 i를 2로 나누어 나머지가 0임을 나타내기 위하여,

[C언어]에서는 $i \% 2 == 0$

답 : $i \% 2 == 0$

PATH 02**1. 데이터베이스 트랜잭션(Transaction)**

- 하나의 작업을 수행하기 위해 필요한 연산들의 집합
- 데이터베이스에서 논리적인 작업의 단위
- **트랜잭션의 특성(ACID)**

원자성	일관성	독립성	지속성
Atomicity	Consistency	Isolation	Durability

답 : 원자성

PATH 03**1. 워크 스루(walk-through)**

코드의 품질을 평가하고, 개선하기 위한 목적으로 수행되는 검토 기법으로 검토회의라고도 하며 프로그램이나 설계서의 오류를 검토과정에서 발견하기 위하여 시행하는 회의 오류의 초기 검출을 목적으로 하며 발견된 오류는 문서화시키고 미리 제공된 자료들을 정해진 절차에 따라 평가합니다.

답 : 워크스루/검토회의**PATH 04****1. Windows 단축키**

- 단축키 : Ctrl, Shift, Alt 등의 조합키와 그 외 키를 결합하여, 특정 기능을 빠르게 쓰는 것을 단축키라고 합니다.

단축키	기능
윈도우 키 + [E]	파일 탐색기 실행
윈도우 키 + [F]	검색
윈도우 키 + [D]	바탕화면 표시
윈도우 키 + [M]	모든 창 최소화
윈도우 키 + [Shift + M]	최소화 복구
윈도우 키 + [Pause]	시스템 등록정보
윈도우 키 + [L]	사용자 로그오프
윈도우 키 + [R]	실행창
윈도우 키 + [+], [-]	돋보기 기능(화면 확대, 축소)

답 : 윈도우키 + E

PATH 05**1. 증감연산자 (값을 증가, 감소 시키는 연산자)**

- 전위연산자(++a / --a) : 피연산자 증감 후, 해당 라인 연산 수행
- 후위연산자(a++ / a--) : 해당 라인 연산 수행 후, 피연산자 증감

2. 코드 분석

```
int x=7; int y=7; int result; // x:7  y:7
result = x > y ? ++x : y--; // ++7 : 7--
// 조건 불만족, False(y--)만 인식;
printf("%2d%2d%2d",x, y, result); // 7 6 7
```

답 : 7 6 7**PATH 06****OSI 7 계층**

응용(Application)

표현(Presentation)

세션(Session)

전송(Transport)

네트워크(Network)

데이터링크(Data Link)

물리(Physical)

TCP/IP 4 계층응용계층
(Application)

전송계층(Transport)

인터넷(Internet)

네트워크 액세스
(Network Access)**답 : 인터넷 계층**

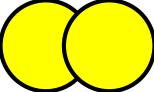
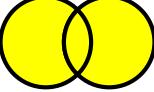
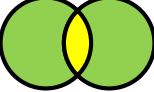
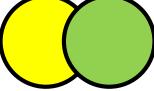
PATH 07

1. MySQL

- SQL을 사용하는 **오픈 소스 관계형 데이터베이스 관리 시스템**
- **오라클 사가** 관리 및 지원
- 다중 사용자 및 다중 스레드 형식을 지원
- C++, java, PHP, Python 스크립트 등을 위한 응용 프로그램 인터페이스(API)를 제공
- 유닉스나 리눅스, 윈도우 운영 체제 등에서 사용 가능

답 : MySQL

PATH 08

집합연산자	형태	정의
UNION (합집합)		중복 행이 제거된 집합
UNION ALL (합집합)		중복 행이 제거되지 않은 집합
INTERSECTI ON (교집합)		두 쿼리 결과에 공통적으로 존재하는 집합
MINUS (차집합)		첫 쿼리에 있고 두번째 쿼리에는 없는 집합

답 : MINUS

PATH 09

프로세스의 상태전이

1. 디스패치(Dispatch Process) : *ready → running*

- 준비 상태에서 실행 상태로 바뀌는 것

2. 보류(Block Process) : *running → blocked*

- 프로세스 실행 중 시간이 다 되기도 전에 입출력 동작을 해야 할 경우
프로세스가 CPU를 반납하고 보류 상태로 들어가는 것

3. 깨움(Wakeup Process) : *blocked → ready*

- 보류 상태 이후 다시 준비 상태로 넘어가는 과정

4. 시간제한(Timer Process) : *running → ready*

- 클락 인터럽트를 통해 프로세스가 일정 시간만 점유할 수 있게 하는 것.

답 : 디스패치(Dispatch)

PATH 10

1. 메소드(Method)

- 반복적인 코드의 사용을 방지하기 위하여, 하나의 항을 만들어서
필요할 때마다 해당 값을 호출하여 반환하는 역할을 수행합니다.

2. 코드 분석

- 메인구문

```
public static void main(String[] args) {
    int num = 0;
    sub(1, 4); ←
    System.out.println(num); // num = 0
} // num에 sub를 대입하지 않아서 출력X
```

호출
num1 = 1
num2 = 4

- 메소드

```
public static int Sum(int num1, int num2){
    int imsub;
    ....
    return imsub;
}
```

답 : 0

PATH 11

1. GRANT : 권한 부여

ex) 이백수에게 회원 테이블 수정 권한 부여

```
GRANT UPDATE ON 회원 TO 이백수
```

2. REVOKE : 권한 해제

ex) 이백수에게 회원 테이블 수정 권한 해제

```
REVOKE UPDATE ON 회원 FROM 이백수
```

답 : FROM

PATH 12

1. CASCADE : 종속

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고
있을 경우 연쇄적으로 함께 변경 또는 삭제합니다.

2. RESTRICT : 참조상태 제한

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고
있을 경우 변경 또는 삭제가 취소됩니다.

답 : CASCADE

PATH 13**1. 구문(Syntax)**

- 데이터의 구조와 형식을 의미하며, 프로토콜 작성에 따라야 하는 문법과 구조입니다.

2. 의미(Semantics)

- 비트의 각 부분을 의미하며, 특정 패턴의 해석과 해석에 기초한 동작을 담당합니다.

3. 시간(Timing)

- 데이터를 전송해야 할 시간의 조정과 전송속도에 대한 영역입니다.

답 : 구문**PATH 14****1. ARP**

- 주소 재해석 및 주소 결정 프로토콜이라 불립니다.
- 논리적인(IP) 주소를 이용해 물리적인(MAC) 주소를 검색합니다.
- 네트워크 계층에 속해있으며, 데이터 링크 계층의 MAC 주소와 연결해 줍니다.

2. RARP

- 역순 주소 결정 프로토콜이라 불립니다.
- 물리적인(MAC) 주소를 이용해 논리적인(IP) 주소를 검색합니다.
- 네트워크 계층에 속해있지만, ARP와는 다르게 데이터 링크 계층의 MAC 주소를 이용해 네트워크 계층의 IP를 검색해 줍니다.

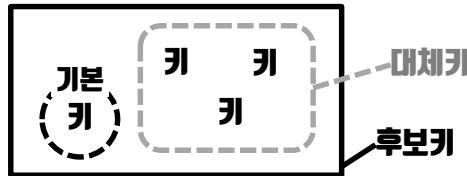
답 : ARP

PATH 15**1. 살충제 패러독스 (Pesticide Paradox)**

- 지속적인 살충제 살포로 내성이 생긴 벌레가 죽지 않는 현상
- 애플리케이션 테스트에서도 동일한 테스트에 대한 비정상적인 결함
검수가 이루어질 수 있습니다. 이러한 현상을 방지하기 위해서는 정기적인 리뷰와 개선이 필요하고, 다양한 시각과 방법으로 테스트 하는 것이 필요합니다.

2. 테스트의 7가지 원칙

- 1) 테스트는 결함이 존재함을 밝히는 활동이다.
- 2) 완전한 테스트는 불가능
- 3) 초기 테스트
- 4) 결함 집중
- 5) 살충제 패러독스
- 6) 테스트는 정황의 의존적이다.
- 7) 오류-부재의 궤변

답 : 살충제 패러독스**PATH 16****키(Key)**

1. 후보키(Key) : 유일성 / 최소성 만족
2. 기본키 : 후보키 중 선정된 키로 중복 값 입력이 불가능하다.
Null 값을 가질 수 없다.
3. 슈퍼키 : 유일성 만족 / 최소성 불만족
4. 대체키 : 기본 키로 선택되지 못한 후보 키
5. 외래키 : 다른 테이블의 행을 식별하는 키

답 : 기본키(Primary Key)

PATH 17

구분	스텝	드라이버
시기	상위 모듈은 있지만 하위 모듈이 없을 때, 하위 모듈로의 테스트를 진행	상위 모듈 없이 하위 모듈만 있을 때, 상위 모듈로의 테스트를 진행
방식	하향식 테스트	상향식 테스트
특징	작성이 쉬운 시험용 모듈	존재하지 않는 상위 모듈간의 인터페이스 역할

답 : 스텝(테스트 스텝)

PATH 18

1. 비트연산자

- 10진수의 정수 값을 2진수로 변환하여 논리 연산을 하는 기능입니다.
- 대표적으로 & , |, ^이 있습니다.
- & (and) : 값이 같은 경우 1을 반환
- | (or) : 값이 둘 중에 하나라도 1이면 1을 반환
- ^ (xor) : 값이 다를 경우 1을 반환

2. 코드분석

```

int a, b, c, result;

a = 12;           // 12 : 0000 1100
b = 5;            // 5 : 0000 0101
                  // 12 & 5(and) : 0000 0100(4)
c = 6;            // 6 : 0000 0110
                  // 4 | 6(or) : 0000 0110 (6)

```

답 : 6

PATH 19**1. 테스트 케이스**

- 특정 요구사항 준수 여부를 확인하기 위해 개발된 입력 값, 실행 조건, 예상된 결과의 집합, 미리 설계하면 오류를 방지할 수 있고 테스트 수행에 필요한 인력, 시간 등의 낭비를 줄일 수 있습니다.

* 테스트 케이스 작성 절차

계획 검토 및 참조 문서 수집
내부 검토 및 우선순위 결정
요구사항 정의
테스트 설계와 방법 결정
테스트 케이스 정의
테스트 케이스 타당성 확인 및 유지보수
테스트 수행

답 : 테스트 케이스

PATH 20**1. UNIX**

- CLI / C언어 기반 / 시분할 온라인 대화식 시스템
- 확장성, 이식성 우수

2. MS-DOS

- CLI / Windows 이전에 사용되던 운영체제

3. 명령어 비교

	UNIX	MS-DOS
속성, 권한 변경	chmod	attrib
목록보기	ls	dir
파일 삭제	rm	del
이름변경, 파일이동	mv	ren

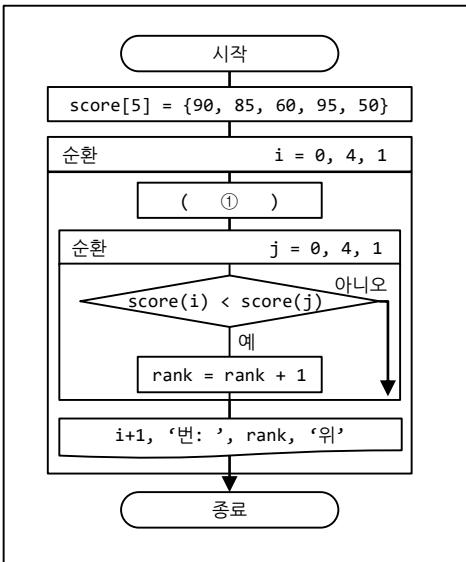
답 : attrib

PATH

**정보처리기능사 실기
예상문제 8회**

01.

배열로 입력 받은 점수의 순위를 출력하는 프로그램을 완성하시오.



02.

다음은 C 언어로 구현된 특정 문자열의 개수를 구하는 프로그램이다. 빈칸에 알맞은 라이브러리 함수를 입력하시오.

```
#include <stdio.h>
#include <string.h>
int main(void) {
    char *str = "Hello";
    printf("%d", ( ① )(str) );
    return 0;
}
```

03. IPv6로 작성된 IP 2001:0230:abcd:ffff::ffff::1111에 사용된 비트(bit) 수를 구하시오.

04. 다음에서 설명하는 네트워크 프로토콜은 무엇인가.

- 신뢰성 있는 전송을 확보한다.
- Flow Control 기능을 수행하여 단계별 데이터 전송 상황을 체크한다.
- 수신 측의 수신 가능상태, 수신 여부 등을 단계별로 체크하며 전달한다.

05. 다음 설명이 나타내는 법칙을 쓰시오.

- 애플리케이션 사용 시간의 80%를 기능의 20%만 사용하기 때문에, 사용 빈도가 높은 기능들에 대해 단순화 및 자동화 한다.
- 애플리케이션의 20%의 잘못된 프로세스만 고치면, 80% 넘는 버그를 제거할 수 있다.

06. 다음 C 언어로 구현된 각 성적에 따른 등급을 나누는 프로그램이다. 빈칸에 알맞은 내용을 입력하시오.

```
#include<stdio.h>
int main()
{
    int grade = 65;

    switch( _____①_____ ){
        case 9 : printf("수"); break;
        case 8 : printf("우"); break;
        case 7 : printf("미"); break;
        case 6 : printf("양"); break;
        _____②_____ : printf("가"); break;
    }
    return 0;
}
```

출력문

양

07.

다음이 설명하는 단어를 적으시오.

인터넷 기반에서 구동되는 컴퓨팅 기술을 의미한다. (①)을 이용하면 응용 프로그램을 필요에 따라 불러 사용하고, 데이터를 손쉽게 저장 및 추출할 수 있다. 태블릿, 휴대폰 등 사용 디바이스가 다양해지면서 이 기술에 대한 중요도가 높아지고 있다.

08.

데이터베이스의 CRUD(Create, Read, Update, Delete)는 각각 SQL명령어로 기본처리를 수행 할 수 있다.
이 중 Read에 해당하는 SQL 명령어는 무엇인가.

- 09.** 다음 C 언어로 구현된 프로그램을 분석하여,
출력 결과를 작성하시오.

```
#include<stdio.h>
int main(){
    int num1, num2, result;
    num1 = num2 = 15;

    result = ++num1 <= num2++;

    printf("%d\n",result);

    return 0;
}
```

- 10.** 연산표기법에는 전위, 중위, 후위 표기법이 있다.
다음의 중위 표기법을 후위 표기법으로 표기하시오.

중위 표기법 : a + (b * c)

11. 다음 C 언어로 구현된 프로그램을 분석하여,
출력 결과를 작성하시오.

```
#include<stdio.h>
int sum(int x, int y);

int main(){
    printf("%d", sum(-10,15) + sum(20,-7));
}

int sum(int x, int y){
    int sum;
    if (x < y)
        return sum = y + x;
    else
        return sum = x + y;
}
```

12. 다음의 설명하는 정의로 알맞은 것은?

(①)는 소프트웨어의 개발 / 유지 보수 중에 발생한 부정 확한 결과를 의미한다. (①)의 원인으로 개발자 실수, 개발 명세서의 잘못된 이해, 서브루틴 기능 오해 등이 있다.

13. 소프트웨어 내부 소스 코드를 테스트하는 기법인 ‘화이트박스테스트’의 테스트 항목인 것을 고르시오.

- 제어 흐름 테스트 · 상태 전이 테스트 · 분기(branch) 테스트
- 경계값 분석 · 경로 테스트 · 데이터 흐름 테스트
- 비교 테스트 · 도메인 테스트

14. DDL은 데이터베이스 구조를 정의하는 언어이다. 다음에서 설명하는 데이터베이스 정의어는?

데이터 정의어(DDL)인 (①)와 (②)는 데이터베이스에 존재하는 오브젝트를 제거할 때 사용한다.
테이블 제거시, (①)는 구조도 남기지 않고 삭제하는 반면, (②) TABLE은 후에도 구조가 남아있다.

15. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include<stdio.h>
main()
{
    int num1, num2, result;
    num1 = 20;
    num2 = 4;

    result = num1 >> 3 > num2;

    printf("%d", result);
}
```

16. 다음 테이블과 쿼리를 참고하여, 결과를 입력하시오.

```
SELECT SUM(급여) FROM 급여표
WHERE 직급 IN (SELECT 직급 FROM 직원 WHERE 근속 > 10;)
```

표1. 직원

이름	직급	근속
김패스	차장	17
이패스	과장	12
최패스	대리	10
박패스	대리	4

표2. 급여표

직급	급여
대리	18000
과장	20000
차장	25000

- 17.** 다음 <점수표> 릴레이션의 차수(Degree)와 카디널리티(Cardinality) 개수를 구하시오.

순번	성명	학과	점수	총점
1	김재연	컴퓨터공학	92	A
2	강민준	시각디자인	80	D
3	나시찬	정보통신	84	C
4	정수한	세무회계	94	A
5	장지웅	경영학과	87	B
6	정현준	사회학과	95	A+

- 18.** E-R Diagram은 개체(Entity)와 개체 관계(Relation)를 도식화한 표현이다. 다음 기호가 의미하는 것은?

기호	의미
사각형	개체집합
타원	속성
마름모	(①)
선	연결

19. OSI 7계층 중 HTTP, FTP가 속한 계층을 적으시오.(5점)

20. 리눅스(Linux)에서 디렉터리 구조 중,
슈퍼유저 홈 디렉토리는?

1.	rank = 1
2.	Strlen
3.	128bit
4.	TCP
5.	파레토 법칙
6.	① : (grade/10) ② : (default)
7.	클라우드 컴퓨팅
8.	Select
9.	0
10.	a b c * +

11.	18
12.	에러(Error)
13.	제어 흐름 테스트/분기 테스트 경로 테스트/데이터 흐름 테스트
14.	① DROP ② TRUNCATE
15.	0
16.	45000
17.	차수(Degree) : 5개 카디널리티(Cardinality) : 6개
18.	관계(Relation)
19.	응용계층
20.	root

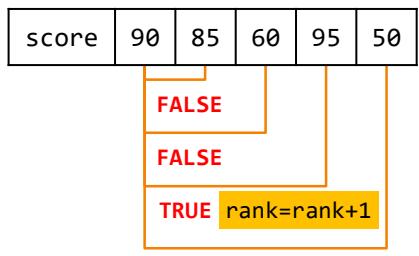
PATH 01**1. 순위 알고리즘**

- 순위의 **초기값은 1**입니다.
- 순위 선정시, 값을 비교하여 작을 경우 순위를 1씩 증가 시킵니다.

ex) 배열을 이용하여 순위 구하기

rank = 1 +1 ▶ score(0):90점의 순위는 2위

score(0) < score(1)



답 : rank = 1

PATH 02**1. 헤더(header)파일**

- 헤더(header)파일이란 특정 기능을 가진 함수의 모음.
- 특정 값을 구하기 위해서는 해당 기능을 지원하는 함수를 가진 헤더파일을 사용해야 합니다.

2. #include <string.h>

- 문자열에 관련된 라이브러리 함수를 가진 헤더 파일입니다.

대표적으로 자주 사용되는 함수

- **strlen** (문자열) - **문자열의 길이 출력**
- **strcmp** (비교할 문자열1, 비교할 문자열2) - **문자열 비교**
- **strcpy** (값을 저장할 공간, 문자열 or [배열]) - **문자열 복사하여 저장**
- **strcat** (값을 저장할 공간, 문자열 or [배열]) - **문자열 이어붙이기**

답 : strlen

PATH 03

구분	IPv4	IPv6
주소 길이	32bit	128bit
표시 방법	8비트씩 4부분 (10진수)	16비트씩 8부분 (16진수)
주소 개수	약 43억 개	무한대에 가깝게 사용 가능
주소 할당	A,B,C 등 클래스 단위의 비순차적 할당	네트워크 규모 및 단말기 수에 따른 순차적 할당
품질 제어	지원 수단 없음	등급별, 서비스별로 패킷 구분
보안 기능	Ipsec 프로토콜 별도 설치	확장 기능에서 기본으로 제공
PnP	지원 수단 없음	지원 수단 있음

답 : 128bit

PATH 04

1. TCP

- 신뢰성 높은 전송방식(CRC 체크, 재전송 기능)
- 단계별 데이터 전송 상황 체크.
- 대표서비스 : FTP, Telnet, Http, SMTP, POP, IMAP 등

2. UDP

- 신뢰성 낮은 전송방식(수신 여부 확인불가)
- 하나의 송신 정보를 다수의 인원이 수신해야 할 경우 사용
- 대표서비스 : SNMP, DNS, TFTP, NFS, NETBIOS 등

답 : TCP

PATH 05

1. 파레토 법칙(80/20법칙)

- 빌프레도 파테로가 '이탈리아 인구의 20%가 이탈리아 전체 부의 80%를 가지고 있다'는 발언에서 시작한 법칙
- 현대에 들어서 애플리케이션 테스트에도 적용중인 법칙
- 발견된 80%의 오류는 20% 모듈에서 발견되므로, 20%의 모듈을 집중적으로 테스트합니다.
ex) RISC 프로세서와 80/20 법칙
- RISC 프로세서는 칩의 사이즈와 비용을 낮추기 위해, 중요한 20%를 최대한 활용하고 80%를 제거하였습니다.

답 : 파레토 법칙

PATH 06

1. switch case문

- 변수의 값을 확인하여, case의 값과 일치하게 되면 해당 실행문을 인식합니다.

구문	해석
<pre>switch(변수){ case 값 : 실행문; break; default : 실행문; break; }</pre>	<pre>switch(grade/10){ // 6 출력 case 9 : printf("수"); break; case 8 : printf("우"); break; case 7 : printf("미"); break; case 6 : printf("양"); break; // 변수의 값과 일치, 양을 출력 default : printf("가"); break; // default란 if의 else와 같음 // case 조건에 부합하지 않으면 출력 }</pre>

답 : ① : (grade/10) ② : (default)

PATH 07**1. 클라우드/클라우드 컴퓨팅**

- 인터넷 기반 컴퓨팅의 일종
- 클라우드(인터넷)에 연결된 다른 컴퓨터로 정보를 처리하는 기술.

2. 대표적 클라우드 서비스**a. IaaS(Infrastructure as a Service)**

- Google, Micro-Soft, Amazon 등에서 제공하는 인프라를 임대하여 이용할 수 있는 서비스

b. PaaS(Platform as a Service)

- 운영체제가 이미 구성되어 있는 상태에서 사용자가 직접 관리할 수 있는 서비스

c. SaaS(Software as a Service)

- 소프트웨어 까지 갖추어져 있어서, 웹 상의 로그인만으로 사용할 수 있는 서비스

답 : 클라우드 컴퓨팅**PATH 08**

기본 처리	SQL	설명
Create	Insert	테이블 내 칼럼에 데이터 추가
Read	Select	테이블 내 칼럼에 저장된 데이터 로드
Update	Update	테이블 내 칼럼에 저장된 데이터 수정.
Delete	Delete	테이블 내 칼럼에 저장된 데이터 삭제.

답 : Select

PATH 09

1. 증감연산자 (값을 증가, 감소 시키는 연산자)

- 전위연산자(`++a` / `--a`) : 피연산자 증감 후, 해당 라인 연산 수행

```
1 ex) int a = 5; int b = 5;
2     printf("%d", ++a, --b) // 결과 : 6 4
```

- 후위연산자(`a++` / `a--`) : 해당 라인 연산 수행 후, 피연산자 증감

```
1 ex) int a = 5; int b = 5;
2     printf("%d", a++, b--) // 결과 : 5 5
3     pinrtf("%d", a, b) // 결과 : 6 4
```

2. 코드 분석

```
int main(){
    int num1, num2, result; // 변수 선언
    num1 = num2 = 15; // num1과 num2에 15 저장
    result = ++num1 <= num2++; // ++15 <= 15--
    // ++15 => 16으로 증가(전위), 15-- => 15 (후위)
    printf("%d\n", result); // 0(False) 출력
```

답 : 0

PATH 10

종류	설명	$a + (b * c)$ 표현 예시
전위 (Prefix)	연산자를 먼저 쓰고, 피연산자를 뒤에 쓰는 표기법	$+ a * b c$
중위 (Infix)	실생활에 사용되는 방식과 동일하게 쓰는 표기법	$a + (b * c)$
후위 (Postfix)	피연산자를 먼저 쓰고, 연산자를 맨 뒤에 쓰는 표기법	$a b c * +$

답 : a b c * +

PATH 11**1. 함수 (≒ JAVA : 메소드)**

- 코드의 사용을 방지하기 위하여, 하나의 항을 만들어서 필요할 때마다 해당 값을 호출하여 반복하는 역할을 수행합니다.

구문	예시
자료형 함수명 (매개변수){ 실행문;}	int function(int num1, num2){ printf("%d %d", num1, num2);}

2. 코드 분석

```
int main(){
    printf("%d", sum(-10,15) + sum(20,-7));
}

int sum(int x, int y){ // 매개변수 x,y 선언 ← 호출
    int sum; // 값을 저장할 공간 선언
    if (x < y) // x가 y보다 작을 시에 부합
        return sum = y + x; // y를 앞으로, x를 뒤로 이동시키어 계산
    else // if에 해당하지 않는 모든 조건
        return sum = x + y; } // x+y를 sum에 저장
```

답 : 18**PATH 12****1. 에러(Error)**

- 소프트웨어 개발 / 유지 보수 중에 발생한 부정확한 결과
- 개발자 실수, 개발 명세서의 잘못된 이해, 서브루틴 기능 오해 등

2. 오류(Fault)

- 프로그램 코드 상에 존재하는 것
- 정상/비정상적인 프로그램 버전간의 차이로 인하여 발생

3. 실패(Failure)

- 정상/비정상적인 프로그램 버전간의 실행 결과의 차이
- 실제 실행 결과를 개발 명세서의 예상 결과와 비교함으로써 발견

4. 결함(Defect)

- 버그, 에러, 오류, 실패, 프로그램 실행에 대한 문제점, 프로그램 개선 사항 등의 전체를 포괄하는 용어이다.

답 : 에러(Error)

PATH 13**1. 화이트 박스 검사(White-Box Testing)**

- 응용 프로그램의 내부 구조와 동작을 검사하는 테스트입니다.
- 즉 개발자가 내부 소스의 코드를 검사하는 작업을 하게 되며, 동작의 작동 유무와 코드까지도 확인하여 테스트할 수 있습니다.
- 종류 : 제어 흐름 테스트, 데이터 흐름 테스트, 분기 테스트, 경로 테스트

2. 블랙박스 검사(Black Box Testing)

- 종류 : 동등 분할, 경계값 분석, 도메인 테스트, 상태 전이 테스트, 페어와이즈 조합 테스트, 결정 테이블

**답 : 제어 흐름 테스트 / 분기 테스트
경로 테스트 / 데이터 흐름 테스트**

PATH 14**1. DDL(Data Definition Language)**

- 데이터베이스 구조를 정의하는 언어이며, 데이터를 생성(CREATE), 수정(ALTER), 삭제(DROP)하는 등 데이터 전체의 구조를 결정하는 역할을 하는 언어입니다.

종류	구분	역할
CREATE	생성	객체(Object) 생성
ALTER	수정	객체 수정
DROP	삭제	객체 삭제 / 구조도 삭제
TRUNCATE		객체 내용 삭제 (초기화)

답 : ① DROP ② TRUNCATE

PATH 15**1. 비트 연산자(시프트 연산자)**

- 해당 대상의 비트(bit)의 위치를 이동시키는 연산자를 의미합니다.

종류	의미
대상 << 이동값(Left)	이동 값만큼 대상의 비트의 자릿수를 좌측으로 이동
대상 >> 이동값(Right)	이동 값만큼 대상의 비트의 자릿수를 우측으로 이동

2. 연산자 우선 순위

우선 순위	연산자
1	산술연산자 [*, /, % -> +, -]
2	시프트 연산자 [>>, <<]
3	비교 연산자 [>, <, >=, <=, ==, !=]

3. 코드 분석

```
result = num1 >> 3 > num2; // ( 0001 0100 >> 3 ) > 4
// 0000 0010(2) > 4 => 0(False) 출력
```

답 : 0**PATH 16****1. 문제해설**

```
SELECT SUM(급여) FROM 급여표
WHERE 직급 IN (SELECT 직급 FROM 직원 WHERE 근속 > 10)
```

표1. 직원

이름	직급	근속
김패스	차장	17
이패스	과장	12
최패스	대리	10
박패스	대리	4

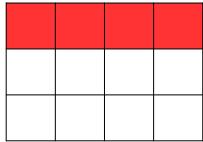
표2. 급여표

직급	급여
대리	18000
과장	20000
차장	25000

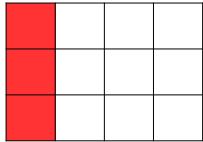
답 : 45000

PATH 17

1. 속성(Attribute)



2. 튜플(Tuple)



릴레이션의 속성(Attribute)의 개수를 차수(Degree)라 하며,
튜플(Tuple)의 개수를 카디널리티(Cardinality)라 부릅니다.

본 <점수표> 테이블의 속성은 5개이므로 차수는 5개 입니다.
첫 행을 제외한 튜플의 수는 총 6줄이므로 카디널리티는 6개 입니다.

**답 : 차수(Degree) : 5개
카디널리티(Cardinality) : 6개**

PATH 18

기호	의미
□	개체집합(Entity)
○	속성 (Attribute)
◇	관계(Relation)
—	연결(link)

답 : 관계(Relation)

PATH 19

1. OSI 7 계층



2. 응용계층

사용자가 네트워크에 접근할 수 있도록 해, 인터페이스나 데이터베이스 관리 등의 서비스를 제공합니다.

3. 전송계층 대표 프로토콜

a. HTTP(HyperText Transfer Protocol)

- 텍스트 기반의 통신 규약으로 웹 상에서 정보를 주고 받는 프로토콜

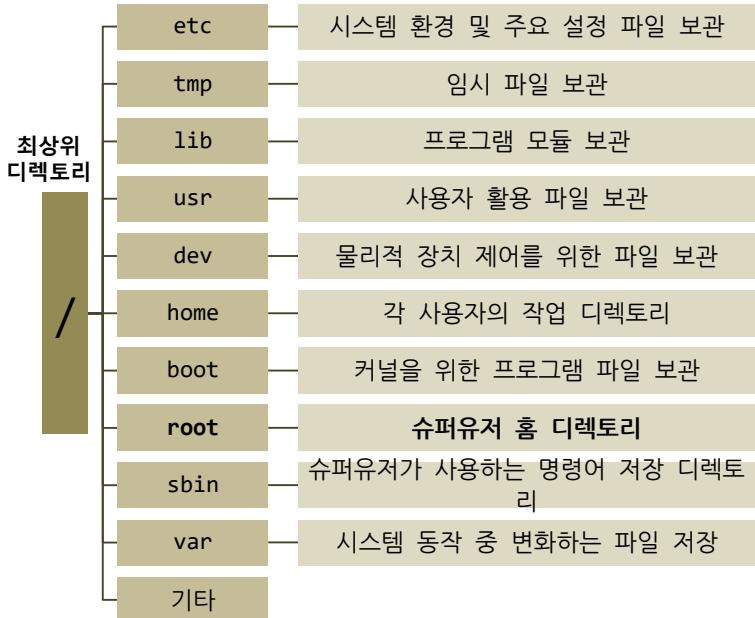
b. FTP(File Transfer Protocol)

- 인터넷 상에서 파일을 주고 받는 프로토콜

답 : 응용 계층

PATH 20

<리눅스 디렉토리 구조>



답 : root

PATH

**정보처리기능사 실기
예상문제 9회**

- 01.** 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class Ideone
{
    public static void main (String[] args{
        int a = 15;
        int b = 4;
        int c = 7;
        int result1 = a ^ b;
        int result2 = a ^ c;

        System.out.println("result1 : " +result1);
        System.out.println("result2 : " +result2);
    }
}
```

- 02.** C언어를 발전시킨 언어로서, 객체 지향적인 특성을 지니며 .NET 환경에서만 실행이 되기 때문에, 다른 운영체제에서는 사용할 수 없는 것은 무엇인가?

- 03.** Linux 서버 등에서 사용하는 데이터베이스의 일종이다.
높은 예산과 복잡한 비즈니스의 고객을 위해 설계되어,
보다 고급 기능 등을 지원하며, 사후처리 시스템이 설비
되어 있는 시스템은?

- 04.** 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int a = 15;
    int b = 4;

    printf("%d\n", ~a);
    printf("%d\n", ~b);
    return 0;
}
```

05.

다음 C 언어로 구현된 “Hello PATH”가 출력되는 프로그램이다. 빈칸에 알맞은 내용을 작성하시오.

```
#include <stdio.h>
#include <string.h>
main(void) {
    char a[15] = "PATH";
    char b[15] = "Hello";
    char c[15] = {0, };

    _____ (c, a);
    _____ (a, b);
    _____ (b, c);

    printf("%s %s", a, b);
}
```

06.

다음 보기의 빈칸에 알맞은 내용을 작성하시오.

- (①)는 2개 이상의 서로 다른 종류의 네트워크를 상호 접속하여 정보를 주고 받을 수 있게 하는 장치이다.
- (②)는 패킷이 목적지로 전송하도록 도와주게 되며 여러 LAN을 연결하여 규모가 큰 네트워크로 확장하기 위해 사용한다.
- (③)는 (②)의 확장 개념으로, 하드웨어를 이용하여 처리하게 되어 속도가 빠르며, 많은 포트 수와 서로 다른 속도로 연결하여 줄 수 있습니다.
- (④)란 네트워크 계층에서 동작하며, 서로 다른 구조와 망의 통신을 위해 사용한다.
- (⑤)는 네트워크의 전송 거리를 연장하기 위해 사용한다.

07.

다음 설명이 나타내는 용어를 쓰시오

- 데이터를 빠르게 찾을 수 있는 수단으로서, 테이블에 대한 조회 속도를 높여 주는 자료구조이다.
- 특정 칼럼 값을 가지고 있는 열 혹은 값을 빠르게 찾기 위해 사용된다.

08.

계속적으로 동기화를 이루지 않아, 사용하지 않는 슬롯을 다른 사람이 사용할 수 있도록 하여 효율성을 높이게 되며, 사용하기 위해서는 하나의 데이터를 보내 연결하게 되며, 이후 실제 데이터를 전송하는 방식은?

09.

다음 설명이 나타내는 용어를 쓰시오.

- 사물 또는 사건으로 정의되며 개체라고도 한다.
- ERD에서 사각형으로 나타낸다.

10.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>
int main(void) {
    float a = 5.124;
    float result;

    result = --a;

    printf("%.3f", result);
    return 0;
}
```

- 11.** 사원 테이블에서 부서번호가 비어있지 않은 사원번호와 이름을 검색하는 SQL문을 작성하시오.

- 12.** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>
int main(void){
    int i;
    int j=5;
    for(i=1; i<=10; i++){
    }
    printf("%d", i-j);
    return 0;
}
```

13. 다음 설명이 나타내는 테스트 수행 방법을 쓰시오.

- 모든 모듈을 동시에 통합 후 테스트 수행
- 드라이버/스텝없이 실제 모듈로 테스트
- 단시간 테스트 가능

14. 사용자 전화 및 사용자 로그오프를 실행하려 할때,
Windows 의 단축키는 무엇인가?

- 15.** 테이블을 구조를 변경하면서 컬럼을 삭제하기 위해 다음 빈칸에 들어갈 명령어는?

```
ALTER TABLE 테이블명 ( ① ) COLUMN 컬럼명
```

- 16.** 윈도우 운영체제에서 레지스트리 편집기를 실행하는 명령어를 쓰시오.

17. 다음의 설명하는 용어를 쓰시오.

18. 데이터베이스 관리시스템(DBMS) 특징 5가지를 쓰시오.

주어진 테스트 케이스에 의해 수행되는 소프트웨어의 테스트 범위
를 측정하는 테스트 품질 측정 기준이며, 테스트의 정확성과 신뢰
성을 향상시키는 역할을 테스트 (①) 라고 한다.

19.

형상 관리 도구 중 다음에서 설명하는 종류를 쓰시오.

초창기에 개발하여 가장 오래된 형상 관리 도구 중 하나로 서버는 단순한 명령 구조를 가진 장점이 있지만,
텍스트 기반의 코드만 지원하는 단점이 있다.

20.

다음은 2진수의 값을 2의보수로 변환하는 프로그램이다.
알고리즘의 빈칸과 프로그래밍 코드의 실행 결과를 쓰시오.

a[5]={1,0,1,0,0}
cp1[5],cp2[2], p , i
c = (①)

i = 0, 4, 1

cp1[i] = 1 - a[i]

p = 0, 4, 1

cp2[4-p] = cp1[4-p] + c;
cp2[4-p] = cp2[4-p] % 2;
c = cp1[4-p] * c;

```
int a[5]={1,0,1,0,0}
int cp1[5],cp2[5],p,i;
int c=( ① );
for(i=0;i<=4;i++)
{
    cp1[i]=1-a[i];
}
for(p=0;p<=4;p++)
{
    cp2[4-p] = cp1[4-p] + c;
    cp2[4-p] = cp2[4-p] % 2;
    c = cp1[4-p] * c;
}
for(i=0;i<5;i++)
{
    printf("%d ",cp2[i]);
}
```

1.	result1 : 11 result2 : 8
2.	C#
3.	오라클(Oracle) DB
4.	-16 -5
5.	Strcpy
6.	게이트웨이, 브리지, 스위치 라우터, 리피터
7.	인덱스(Index)
8.	ATM(Asynchronous Transfer Mode)
9.	엔티티(Entity)
10.	4.124

11.	SELECT 사원번호, 이름 FROM 직원 WHERE 부서번호 IS NOT NULL;
12.	6
13.	빅뱅(Big Bang)
14.	원도우키 + L
15.	DROP
16.	Regedit
17.	커버리지
18.	무결성, 일관성, 회복성, 보안성 효율성
19.	CVS(Concurrent Versions System)
20.	① 1 결과 : 0 1 1 0 0

PATH 01**1. 비트 연산자(이항 연산자)**

- 피연산자를 10진수의 정수 값을 2진수로 변환하여 논리 연산을 하는 기능

ex) 1(십진수) -> 0000 0001(이진수), 2(십진수) -> 0000 0010
 3(십진수) -> 0000 0011(이진수)

종류	예시	의미
& (and)	int a = 5; int b = 2; printf("%d", a&b);	a = 0101 b = 0010 → 0000(0) a와 b 값이 모두 1이면 1 출력
 (or)	int a = 5; int b = 2; printf("%d", a b);	a = 0101 b = 0010 → 0111(7) 둘 중에 하나라도 1이면 1 출력
^ (xor)	int a = 5; int b = 3; printf("%d", a^b);	a = 0101 b = 0011 → 0110(6) a와 b의 값이 다르면 1 출력

답 : result1 : 11 result2 : 8

PATH 02**1. C언어**

- 절차지향(구조적) 언어
- 어떤 플랫폼에서도 코딩 가능 (ex: 리눅스, 유닉스, 윈도우)
- 메모리 직접 관리 가능

2. C++

- C언어를 기반으로 만들어짐
- 객체지향적 언어
- 어떤 플랫폼에서도 코딩 가능
- 메모리 직접 관리 가능

3. C#

- C언어, C++, JAVA 등의 장점을 기반으로 만들어짐
- 객체지향적 언어
- NET환경에서 사용되기 때문에, 윈도우 OS만 사용 가능
- 자동으로 누수되는 메모리 자원을 회수

답 : C#

PATH 03**1. 오라클(Oracle) DB**

- 리눅스(Linux) 서버 등에 자주 이용(Windows는 MS SQL 사용)
- 높은 구매 비용 및 사후 처리 시스템 설비
- 중앙 집중 방식의 대용량의 데이터베이스에 적합한 성능과 기능을 지원

2. MySql

- 리눅스(Linux) 서버 등에 자주 이용
- 낮은 구매 비용 및 무료 GUI 등을 지원하기도 함.
- 매우 적은 오버헤드 사용
- 타 DBMS 시스템에 비해 사용하기가 쉽다.
- 고급 기능 등을 지원하기 시작

답 : 오라클(Oracle) DB

PATH 04**1. 보수(NOT) 연산자**

- 보수(Complement)란 비트(Bit)를 통해 표현할 수 있는 것이 양수로 한정되기에, 음수를 표현하기 위해 사용하며, C언어의 경우 2의 보수를 출력한다.

1의 보수 계산 : 0을 1로, 1을 0으로 반전

ex) 0000 1100(+12) → 1111 0011(-12)

// 비트의 가장 처음 값은 부호의 역할 (0 = 양수, 1 = 음수)

2의 보수 계산 : 1의 보수를 보완하기 위해 기존의 값에 1을 더한 값

ex) 1111 0011(-12) + 1 → (-12+1) = -13

// TIP : 답을 편하게 구하기 위해서 정보처리기능사 용도로 개편하여
실제 계산 방법과는 다름을 공지드립니다.

종류	의미
~ 값	보수(NOT) 연산 작업

**답 : -16
-5**

PATH 05

1. 헤더(header)파일

- 헤더(header)파일이란 특정 기능을 가진 함수의 모음.
- 특정 기능을 이용한 값을 구하기 위해서는 해당 함수를 지원하는 헤더파일을 필요

헤더 파일	자주 사용하는 함수
#include<string.h> (문자열 관련)	strlen (문자열) // 문자열 갯수 출력 strcat (불일 대상, 불일 문자) // 문자열 이어주기 strcpy (저장 변수, 복사할 문자) // 문자열 복사 strtok (문자열, 구분자) // 문자열 자르기

답 : **strcpy**

PATH 06

1. 허브(Hub)

- 다수의 장치들을 하나로 연결하는 장치이며, 연결된 장치끼리 상호 간의 통신을 할 수 있는 장비

2. 리피터(Repeater)

- 허브 등으로 연결하여 장거리로 통신할 경우 감쇠된 신호를 수신하여 증폭한 후, 다음 구간으로 재전송

3. 브리지(Bridge)

- 패킷의 목적지 주소를 검사할 수 있고 소프트웨어적인 방법을 통해 프레임(데이터)를 목적지 주소를 기준으로 1:1로 연결하여 전송

4. 스위치(Switch)

- 브리지의 확장된 개념으로 기본 기능은 브리지와 동일하지만 하드웨어적 방식을 사용하여, 전송 중 패킷의 충돌이 일어나지 않도록 목적지 주소를 기준으로 해당 포트들과 1:n으로 연결

5. 라우터(Router)

- 네트워크 계층에서 서로 다른 구조의 망을 연결 가능하여, LAN과 MAN, WAN 등을 연결하는데 사용

6. 게이트웨이(Gateway)

- 다른 종류의 네트워크 등을 상호 접속하여 정보를 주고 받을 수 있는 장치

답 : **게이트웨이, 브리지스위치, 라우터, 리피터**

PATH 07**1. 인덱스(Index)의 장점**

- 테이블 검색속도와 정렬 속도가 향상됨
- 인덱스를 기본키로 자동설정 할 수 있어서 관리가 용이함

2. 인덱스(Index)의 단점

- 인덱스는 별도로 만들어야 해서 파일크기가 늘어남
- 인덱스에는 갱신 개념이 없어서 추가로 수행해야 함
- 인덱스를 생성하여야 하므로 초기 작업시간이 많이 소요됨

답 : 인덱스(Index)

PATH 08**1. 비동기식 전송 모드 (Asynchronous Transfer Mode)**

- 수신기 쪽에서 자신의 신호를 사용하여 슬롯에 데이터를 전송하여 연결하게 되며, 실제 데이터를 전송 (송수신 간의 신호 전송)
- 데이터를 일정한 크기로 나누고, 각 패킷의 전송에 별도의 동기 신호 없이 start-bit와 stop-bit를 부여하여 전송

2. 동기식 전송 모드 (Synchronization Transfer Mode)

- 신호를 일정 주기로 나누어, 시분할 다중 전송
- 동기화를 위해서는 수신자와의 신호가 동일하여야 하며, 전송속도가 서로 일정해야 한다.
- 송신에 사용하지 않더라도 시간 슬롯을 할당하기에 회선 사용 효율이 떨어진다.

답 : ATM(Asynchronous Transfer Mode)

PATH 09**1. 엔티티(Entity)**

- 사물 또는 사건으로 정의되며 개체라고도 한다. ERD에서 엔티티는 사격형으로 나타내고 사격형 안에는 엔티티의 이름을 넣는다.

2. 속성(Attribute)

- 엔티티가 가지고 있는 요소 또는 성질을 속성이라 부른다. 속성은 선으로 연결된 동그라미로 표기하거나 표형식으로 표기하기도 한다.

3. 관계(Relationship)

- 두 엔티티 간의 관계를 정의한다.

답 : 엔티티(Entity) or 개체

PATH 10**1. 증감연산자 (값을 1씩 증가, 감소 시키는 연산자)**

- 전위연산자(++a / --a) : 피연산자 증감 후, 해당 라인 연산 수행

```
1 ex) int a = 5; int b = 5;  
2     printf("%d", ++a, --b); // 결과 : 6 4  
3     pinrtf("%d", a, b); // 결과 : 6 4
```

- 후위연산자(a++ / a--) : 해당 라인 연산 수행 후, 피연산자 증감

```
1 ex) int a = 5; int b = 5;  
2     printf("%d", a++, b--); // 결과 : 5 5  
3     pinrtf("%d", a, b); // 결과 : 6 4
```

답 : 4.124

PATH 11**1. 널(NULL)**

- 데이터베이스 내의 데이터 값이 존재하지 않는다는 것을 의미
- '값이 부족' 하다나 '0값'이라는 의미와 동일한 의미가 아님에 주의
- 비어 있다 **IS NULL** / 비어있지 않다 **IS NOT NULL**

2. SELECT 구문

- 대표적인 구문의 형식은 아래와 같습니다.

```
SELECT 컬럼
FROM 테이블
WHERE 조건문;
```

**답 : SELECT 사원번호, 이름
 FROM 직원
 WHERE 부서번호 IS NOT NULL;**

PATH 12**1. for문 (횟수 반복)**

- 지정한 횟수만큼 반복적으로 수행하기 위한 구문입니다.
- 증감 연산자의 경우, 반복문이 완료되면 증가됩니다.

```
1 int i; // 외부에서 호출하기 위한 변수 선언
2 for( i=1; i<=5; i++){
3     // 내부 변수 선언 시 외부에서 출력 불가
4     printf("%d\n", i); // 1 ~ 5 출력
5 }
6 printf("%d\n", i); // 6 출력(반복문 외부)
```

답 : 6

PATH 13

1. 통합 테스트

	빅뱅(Big Bang)	상향식(Bottom Up)	하향식(Top Down)
수행방법	모든 모듈을 동시에 통합후 수행	최하위 모듈부터 점진적으로 상위 모듈과 함께 수행	최상위 모듈부터 하위 모듈들을 통합하며 수행
드라이버/ 스텝	X	드라이버 필요	스텝 필요
장점	단시간 테스트 가능 작은 시스템에 유리	장애 위치 파악 쉬움 모듈 개발 시간 낭비가 없음	장애 위치 파악 쉬움 이론 프로토타입 가능 중요 모듈의 선 테스트 가능 결함 조기 발견 가능
단점	장애 위치 파악 어려움 모든 모듈 개발	이론 프로토타입 어려움 중요 모듈이 마지막으로 테스트될 가능성 높음	많은 스텝이 필요 하위 모듈들의 불충분한 테스트 수행

답 : 빅뱅(Big Bang)

PATH 14

1. Windows 단축키

- 단축키 : Ctrl, Shift, Alt 등의 조합키와 그 외 키를 결합하여, 특정 기능을 빠르게 쓰는 것을 단축키라고 합니다.

단축키	기능
윈도우 키 + [L]	사용자 로그오프
윈도우 키 + [F]	검색
윈도우 키 + [D]	바탕화면 표시
윈도우 키 + [M]	모든 창 최소화
윈도우 키 + [Shift + M]	최소화 복구
윈도우 키 + [Pause]	시스템 등록정보
윈도우 키 + [E]	파일 탐색기 실행
윈도우 키 + [R]	실행창
윈도우 키 + [+], [-]	돌보기 기능(화면 확대, 축소)

답 : 윈도우 키 + L

PATH 15**1. ALTER TABLE**

- ADD : 컬럼 추가

```
ALTER TABLE 테이블명
ADD 컬럼명 데이터 유형;
```

- DROP COLUMN : 컬럼 삭제

```
ALTER TABLE 테이블명
DROP COLUMN 컬럼명;
```

- MODIFY COLUMN : 제약조건 변경

```
ALTER TABLE 테이블명
MODIFY 컬럼명 데이터 유형;
```

- RENAME COLUMN : 컬럼명 변경

```
ALTER TABLE 테이블명
RENAME COLUMN 컬럼명 TO 새로운컬럼명
```

답 : DROP

PATH 16**1. 레지스트리 (Registry)**

- 프로세스종류, 주기억장치의 용량, 주변장치의 정보, 시스템 매개변수, 응용소프트웨어에서 취급하는 파일 타입과 매개변수 등을 기록한 저장소이다.
- 모든 하드웨어, 운영 체제 소프트웨어, 대부분의 비운영 체제 소프트웨어, 사용자 PC 선호도 등에 대한 정보와 설정이 들어 있다

명령어	기능
regedit	레지스트리 편집기
msconfig	시스템 구성 유틸리티
cmd	도스명령프롬프트
taskmgr	작업관리자
netplwiz	사용자 권한 설정
services.msc	서비스 관리자

답 : regedit

PATH 17

구분	의미
기능 기반	전체 기능을 모수로 설정하고 실제 수행된 기능의 수를 측정하는 방법, 100% 달성을 목표로 하며, 화면 수를 모수로 사용할 수도 있다.
라인	전체 소스 코드의 Line 수를 모수로 측정하는 방법 단위테스트에서는 라인 커버리지를 척도로 삼기도 한다.
구문	코드 구조 내의 모든 구문에 대해 한 번 이상 수행하는 테스트 커버리지
조건	결정 포인트 내의 모든 개별 조건식에 대해 수행하는 테스트 커버리지
결정	모든 분기문에 대해 수행하는 테스트 커버리지
변형 조건/ 결정	조건과 결정을 복합적으로 고려한 측정 방법으로 다른 개별적인 조건식의 결과에 상관없이 독립적으로 전체 조건식의 결과에 영향을 주는 테스트 커버리지

답 : 커버리지**PATH 18**

특징	설명
무결성	부적절한 자료가 입력되어 동일한 내용에 대하여 서로 다른 데이터가 저장되는 것을 허용하지 않는 성질
일관성	삽입, 삭제, 갱신, 생성 후에도 저장된 데이터가 변함없이 일정
회복성	장애가 발생하였을 시 특정 상태로 복구되어야 하는 성질
보안성	불법적인 노출, 변경, 손실로부터 보호되어야 하는 성질
효율성	응답 시간, 저장 공간 활용 등이 최적화되어 사용자, 소프트웨어, 시스템 등의 요구 조건을 만족 시켜야 하는 성질

답 : 무결성, 일관성, 회복성, 보안성, 효율성

PATH 19

형상관리도구

- 형상 식별, 버전 관리, 변경 통제, 형상 감사, 상태 보고 등 형상을 식별하고 관리하는 도구

도구	설명
CVS	가장 오래된 형상관리 도구 중 하나로 서버는 단순한 명령 구조를 가진 장점이 있지만 텍스트 기반의 코드만 지원하는 단점이 있다.
SVN	CVS의 단점을 보완해 현재 가장 대중화된 도구 중 하나로 다양한 GUI 도구가 존재하고 압축을 통해 서버의 공간을 절약하는 장점을 가지고 있다.
Git	리눅스의 커널 개발을 위해 만든 형상 관리 시스템이다. CVS와 SVN의 단점을 모두 보완하는 장점은 있으나 중앙집중형이 아닌 분산형 방식으로 스스로 저장공간이 필요하며 개념이 다르므로 개발자에게 학습할 시간이 필요하다.

답 : CVS(Concurrent Versions System)

PATH 20

1. 배열을 이용한 2진수의 2의보수화

- 설계 방법

2진수의 2의보수는 1의보수 변경후 맨 뒷자리에 1을 더하여 연산

2. 2진수 10100 의 2의보수화

```

int a[5]={1,0,1,0,0} // 비트 및 임시공간 변수 선언
int cp1[5],cp2[5],p , i;
int c = 1; // 2의 보수를 위한 변수 초기값
for(i=0;i<=4;i++)
{
    cp1[i]=1-a[i]; // 1의보수 : 1 - 입력값
}                                a[5] : 1 0 1 0 0 → cp1[5] : 0 1 0 1 1
for(p=0;p<=4;p++)
{
    cp2[4-p] = cp1[4-p] + c; // 고정수를 이용한 후순위 연산
    cp2[4-p] = cp2[4-p] % 2; // 2진수 연산을 위한 나머지연산
    c = cp1[4-p] * c;           0,1 = 변경없음 / 2 → 0
}

```

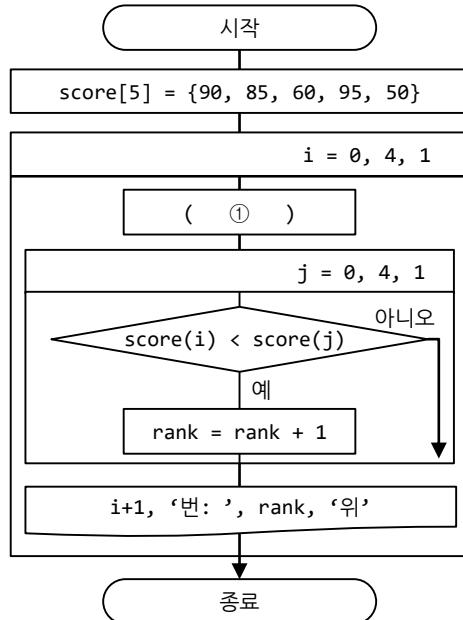
답 : ① 1 결과 : 01100

PATH

**정보처리기능사 실기
예상문제 10회**

01.

배열로 입력 받은 점수의 순위를 출력하는 프로그램을 완성하시오.



02.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

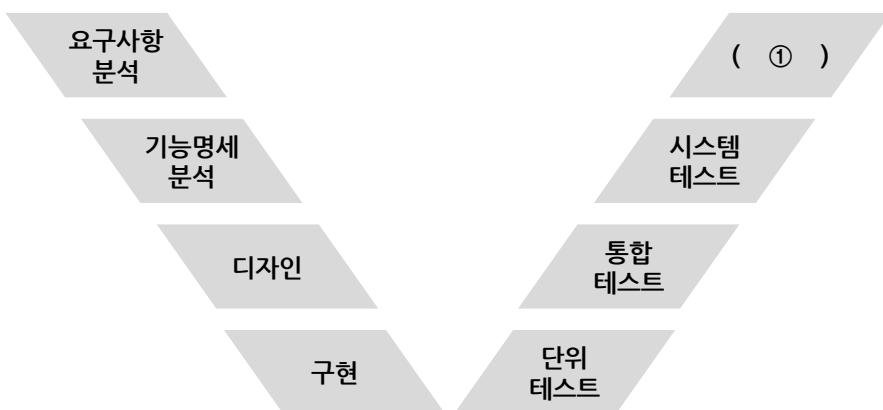
int main(void) {
    int num = 10;
    int result;

    num << 2;
    result = num >> 2;

    printf("%d", result);
    return 0;
}
```

03. <보기>를 참고하여, DCL에 관련된 명령어를 작성하시오.

COMMIT, ROLLBACK, SELECT, DROP,
CREATE, ALTER, DELETE, UPDATE

04. 다음 소프트웨어 생명 주기의 V-모델의 빈칸으로
알맞은 것은?

05. 소프트웨어에 발생한 결함을 체계적으로 관리할 수 있도록 도와주는 도구를 결함 관리 도구라고 한다 다음에서 설명하는 결함 관리 도구는 무엇인가?

- 결함 및 이슈 관리 도구로 소프트웨어 설계 시 단위별 작업 내용을 기록할 수 있어 결함 추적도 가능한 도구

06. 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int num1 = 1;
    int num2 = 9;
    int re;
    re = num1>num2 && num1==num2 ? num1 : num2;

    printf("%d", re);
    return 0;
}
```

07.

다음은 Windows의 ipconfig와 같이 UNIX 기반 운영체제에서 현재 네트워크의 인터페이스 설정 등을 확인할 수 있도록 도와주는 명령어는?

08.

다음은 컴퓨터에 관련된 설명이다.
빈칸에 알맞은 내용을 완성하시오.

1. (①)란 하드웨어와 사용자 사이에 존재하여, 사용자가 하드웨어의 자원을 편하게 사용하기 위해 만든 플랫폼을 의미한다.
2. 하드웨어와 소프트웨어의 중간 형태를 가진 펌웨어의 일종으로, (①)를 도와, 소프트웨어를 통해 하드웨어를 제어할 수 있는 것은 (②)이다.
3. (③)은(는) (②) 등을 저장하여 컴퓨터에게 하드웨어를 제어할 수 있도록 데이터를 제공하나, RAM에 저장되어 있어서 전원 공급을 받지 않으면 데이터가 사라지게 된다..

- 09.** 다음의 A로 시작하는 문자를 찾는 데이터베이스 구문이다.
쿼리의 빈칸을 완성하시오.

autu	price	Rank
1	5.4	A_1
2	3.2	B_1
1	4.2	A_3
1	2.2	C_2
3	1.8	D_3

```
SELECT * FROM Function WHERE RANK (      ) 'A%'
```

- 10.** 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int i = 1;

    while(i<5)
    {
        printf("%d", i);
        i++; }
    printf("%d", i);
    return 0;
}
```

- 11.** 다음 <표>를 참고하여, 차수(Degree)와 카디널리티(Cardinality)의 개수를 구하시오.

name	Price	autu	kind
RG_A123	4.29	4	One piece
RG_B425	5.29	2	Sweater
RG_ML123	5.59	1	Skirt
RG_MB521	2.29	3	Blue jeans

- 12.** 다음의 <보기>를 참고하여, 해당 기능이 무엇인지 작성하시오.

네트워크 장치로부터 자동으로 정보를 수집 및 관리하기 위해 사용하는 프로토콜로서, 실시간으로 네트워크 장비를 모니터링하고 데이터를 변경하여 동작할 수 있도록 합니다. 하지만 관리가 편리한 대신 외부 접속에 대한 취약점 등이 존재

13.

다음의 테이블을 참고하여, 해당 쿼리를 완성하시오.

name	Price	autu	kind
RG_A123	4.29	4	One piece
RG_B425	5.29	2	Sweater
RG_ML123	5.59	1	Skirt
RG_MB521	2.29	3	Blue jeans



name	Price	kind
RG_A123	4.29	One piece
RG_B425	5.29	Sweater
RG_ML123	5.59	Skirt
RG_MB521	2.29	Blue jeans

() TABLE 상품상세 DROP COLUMN autu

14.

테스트 결과가 올바른지 판단하기 위해 사전에 정의된 참 값을 대입하여 비교하는 기법 및 활동을 테스트 오라클이라고 한다. 다음이 설명하는 테스트 오라클의 종류를 쓰시오.

샘플링 오라클의 단점을 개선하기 위해 특정 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 나머지 입력 값들에 대해서는 추정으로 처리하는 오라클

15. 다음이 설명하는 용어를 쓰시오.

- 데이터베이스에 저장된 데이터 값과 그것이 표현하는 현실 세계의 실제값이 일치하는 정확성을 의미
- 부적절한 자료가 입력되어 동일한 내용에 대하여 서로 다른 데이터가 저장되는 것을 허용하지 않는 성질

16. 다음이 설명하는 애플리케이션 성능 테스트 도구를 쓰시오.

HTTP, FTP, LDAP 등 다양한 프로토콜을 지원하는 안전성, 확장성, 부하, 기능 테스트 도구

17. 다음 JAVA 언어로 구현된 “Hello PATH”가 출력되는 프로그램이다. 빈칸에 알맞은 내용을 작성하시오.

```
class Ideone
{
    public static void main (String[] args)
    {
        System.out.println("Hello");
        System.out.printf("PA");
        System.out.print("TH");
    }
}
```

18. 다음 설명이 나타내는 용어를 쓰시오

- 인터넷을 통하여 원격지의 호스트 컴퓨터에 접속할 때에 지원되는 인터넷 표준 프로토콜
- 데이터 전송시 암호화 작업이 없어 보안성이 낮음
- TCP/IP 기반의 프로토콜로 포트 번호는 23번

19.

[보기]를 참고하여, SELECT문의 문법 순서를 완성하시오.

문법 순서 :

SELECT -> () -> () -> GROUP BY ->
HAVING -> ()

[보기]

ORDER BY, WHERE, FROM

20.

다음이 설명하는 용어를 쓰시오.

가장 보편화된 데이터베이스 관리시스템으로
데이터를 저장하는 테이블의 일부를 다른 테이블과
상호 관계로 표시하며 상관관계를 정리하고 변화하는 업무나
데이터 구조에 대한 유연성이 좋아 유지 관리가 용이하다.

1.	rank = 1
2.	2
3.	COMMIT, ROLLBACK
4.	인수테스트 (Acceptance Testing)
5.	Mantis
6.	9
7.	Ifconfig
8.	① : 운영체제(OS) ② : BIOS ③ : CMOS
9.	LIKE
10.	12345

11.	차수 : 4개 카디널리티 : 4개
12.	간이 망 관리 프로토콜(SNMP)
13.	ALTER
14.	추정 오라클 or 휴리스틱 오라클
15.	테이터 무결성
16.	JMeter
17.	Hello PATH
18.	Telnet
19.	FROM, WHERE, ORDER BY
20.	RDBMS

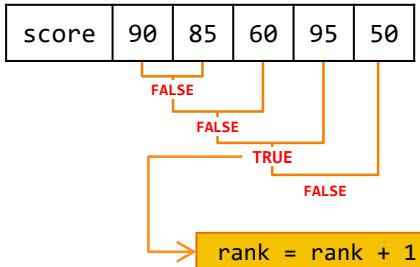
PATH 01**1. 순위**

- 순위의 초기값은 1
- 순위 선정시, 값을 비교하여 작을 경우 순위를 증가(+1)

ex) 배열을 이용하여 순위 구하기

`rank = 1` `+1`

`score(0) < score(1)`



▶ `score(0):90점의 순위는 2위`

답 : rank = 1

PATH 02**1. 비트 연산자(시프트 연산자)**

- 해당 대상의 비트(bit)의 위치를 이동시키는 연산자를 의미합니다.

종류	의미
대상 << 이동값 (Left)	이동 값만큼 대상의 비트의 자릿수를 좌측으로 이동
대상 >> 이동값 (Right)	이동 값만큼 대상의 비트의 자릿수를 우측으로 이동

2. 코드 분석

```

int num = 10; // num는 10 저장
int result;

num << 2; // 값을 저장할 변수 선언 X
result = num >> 2; // 0000 1010
                  → 0000 0010
printf("%d", result); // result는 2 출력
  
```

답 : 2

PATH 03

1. DDL (Data Definition Language)

명령어	의미
CREATE	데이터베이스내의 릴레이션(테이블)을 생성하는 명령어
ALTER	데이터베이스내의 릴레이션(테이블)의 칼럼을 변경하는 명령어
DROP	데이터베이스내의 릴레이션(테이블)을 삭제하는 명령어
TRUNCATE	릴레이션에서 모든 행을 삭제하는 명령어

2. DML (Data Manipulation Language)

SELECT	데이터베이스 내의 데이터를 조회 및 검색하는 명령어
UPDATE	릴레이션 내에 데이터를 수정하는 명령어
DELETE	릴레이션 내에 데이터를 삭제하는 명령어

3. DCL (Data Control Language)

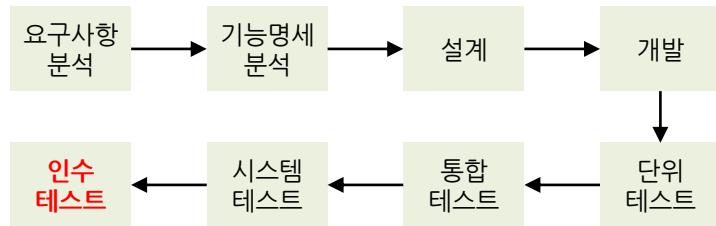
COMMIT	데이터베이스 내의 모든 작업의 완료를 의미하는 명령어
ROLLBACK	문제가 발생할 시에, 처리과정의 변경 사항을 취소하는 명령어
GRANT	계정에 대한 권한을 부여하는 명령어
REVOKE	계정에 대한 권한을 삭제하는 명령어

답 : COMMIT, ROLLBACK

PATH 04

1. V-모델

- 소프트웨어 개발 프로세스인 폭포수 모델의 확장된 형태
- 폭포수 모델과 달리, 왼쪽은 아래로 오른쪽은 위로 진행하며 알파벳 V모양으로 진행 된다.
- 순서



답 : 인수테스트(Acceptance Testing)

PATH 05

구분	의미
Mantis	결함 및 이슈 관리 도구, 소프트웨어 설계 시 단위별 작업 내용을 기록할 수 있어 결함 추적도 가능한 도구
Trac	결함 추적은 물론 결함을 통합하여 관리할 수 있는 도구
Redmine	프로젝트 관리 및 결함 추적이 가능한 도구
Bugzilla	결함 신고, 확인, 처리 등 결함을 지속적으로 관리할 수 있는 도구 결함의 심각도와 우선순위를 지정할 수 있음

답 : Mantis

PATH 06

1. 삼항 연산자

- 조건에 부합할 경우, True와 False에 지정한 값을 출력하는 구문
(# 비교 연산자는 True 혹은 False만 표현)

구문	예시
조건 ? True : False // ? = 조건과 값을 구분 // : = 값1과 값2 구분	int a = 10; int b = 3; char result; // 출력 내용 : 단일문자 result = num1 > num2 ? 'A' : 'B' printf("%c", result); // 'A' 출력

2. 논리 연산자

- 작성 시, 하나가 아닌 두 가지 이상의 조건을 작성하기 위해 사용

구문	의미
조건1 && 조건2 (AND)	조건1과 조건2가 모두 만족할 시에, TRUE를 반환
조건1 조건2 (OR)	조건1과 조건2 중에 하나만 만족하더라도, TRUE를 반환

답 : 9

PATH 07**1. Windows / DOS 의 ipconfig**

- 해당 네트워크 등의 인터페이스를 확인할 수 있는 명령어로서 (= UNIX 기반의 ifconfig)

ex)

```
이더넷 어댑터 이더넷 2:  
연결별 DNS 접미사 . . . . . :  
  흉크로컬 IPv6 주소 . . . . . : fe80::481c:7dfd:9cac:51b8%0  
IPv4 주소 . . . . . : 192.168.1.220  
서브넷 마스크 . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . : 192.168.1.1
```

2. Windows / DOS 의 ipconfig

- 사용자의 계정 및 로그인 정보 등을 확인하는 명령어 (= Unix 기반의 finger)

3. Windows / DOS 의 ping

- 해당 네트워크가 상대와 연결 되어 있는지 확인할 수 있는 명령어 (= UNIX 기반의 ping)

답 : ifconfig**PATH 08****1. 운영체제(Operating System)**

- 사용자가 편리하고 쉽게 하드웨어의 자원을 이용하여 소프트웨어 등을 이용할 수 있도록 도와주는 플랫폼

2. BIOS(Basic Input Output System)

- 기능적으로는 소프트웨어이지만, 하드웨어에 위치하여, 하드웨어의 자원을 이용할 수 있도록 도와줌 (펌웨어)
- 예전에는 ROM에 저장되어 내용을 변경할 수 없었으나 현재에는 오류 및 업데이트 개선 등을 이유로 플래시 메모리에 저장되어 내용을 변경할 수 있음

3. CMOS

- CPU, HDD 등의 장치의 연결 상태를 BIOS에 저장할 수 있음
- RAM에 저장되어 있으며, 메인보드의 수은 전지 탈착 시에 데이터가 손실됨

답 : 운영체제(OS), BIOS, CMOS

PATH 09

1. Where 문

- 특정 조건에 부합하는 값만을 호출하기 위해 사용
- 해당 키워드 뒤에 조건을 입력하는게 보통이나, 조건이 불명확할 경우에 LIKE와 함께 사용하여, 컬럼명을 지정할 수 있음

2. LIKE 문

- Where를 이용한 조건 값이 명확하지 않을 때 사용
- %는 “모든 문자”를 의미하며, _는 “단일 문자”를 의미

답 : LIKE

PATH 10

1. 반복문(While)

- 조건에 부합하면 특정한 내용을 반복적으로 수행하기 위한 구문

ex) `while(조건)`

{ 실행문

} → 조건범위 내에서 반복 실행

2. 코드분석

```
int i = 1; // i는 1 저장

while(i<5) // i가 5 미만일 시 반복
{
    printf("%d", i); // 1, 2, 3, 4 출력
    i++; // i = i + 1
}
printf("%d", i); // i++ 이후, i는 5를 가짐
5 출력
```

답 : 12345

PATH 11

1. 속성(Attribute)

1	2	3	4

2. 튜플(Tuple)

1			
2			
3			

릴레이션(테이블)에서 속성의 개수를 차수(Degree)로 표현하며,
튜플의 개수를 기수 혹은 카디널리티(Cardinality)라 표현합니다.

TIP) 속성은 컬럼 혹은 필드로도 표현합니다.

튜플은 행 혹은 레코드로 불리기도 합니다.

해당 문제의 경우, 속성은 4개이므로, 차수는 4개이며,
첫 행을 제외한 튜플의 개수는 총 4줄이므로, 카디널리티는 4입니다.

답 : 차수(Degree) : 4개

카디널리티(Cardinality) : 4개

PATH 12

1. SNMP (Simple Network Management Protocol)

- 네트워크 장치로부터 정보를 수집 및 관리할 수 있으며, 해당 데이터를 수정하여 장치를 운영할 수 있도록 사용하는 프로토콜
- SGMP에서 간소화 작업을 하였으며, 이를 발전하여 만듬
- 높은 편리성 대신 여러가지 취약점이 존재

답 : 간이 망 관리 프로토콜(SNMP)

PATH 13

DDL (Data Definition Language)

- 데이터 정의 언어로서, 데이터를 저장하기 위한 테이블(릴레이션) 등을 정의하는 언어입니다.
- 대표적으로 CREATE(생성), ALTER(수정), DROP(삭제) 등이 있습니다.

명령어	의미
ALTER	<p>데이터베이스내의 릴레이션(테이블)의 칼럼을 변경하는 명령어</p> <p>ex) 1. ALTER TABLE 테이블명 ADD 컬럼명 데이터타입 ALTER ADD : ALTER문의 열 추가 2. ALTER TABLE 테이블명 MODIFY 컬럼명 데이터타입 ALTER MODIFY : ALTER문의 타입 변경 3. ALTER TABLE 테이블명 DROP 컬럼명 ALTER DROP : ALTER문의 열 삭제</p>

답 : ALTER

PATH 14

종류	정의
참 (True)	모든 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 발생된 모든 오류를 검출할 수 있음
샘플링 (Sampling)	특정한 몇몇 테스트 케이스의 입력 값들에 대해서만 기대하는 결과를 제공하는 오라클
추정 (Heuristic)	샘플링 오라클을 개선한 오라클로, 특정 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 나머지 입력 값들에 대해서는 추정으로 처리하는 오라클
일관성 검사 (consistent)	애플리케이션의 변경이 있을 때, 테스트 케이스의 수행 전과 후의 결과 값이 동일한지를 확인

답 : 추정 오라클 or 휴리스틱 오라클

PATH 15**1. 무결성 종류**

- 개체 무결성
- 릴레이션에서 기본키를 구성하는 속성을 널(NULL)값이나 중복값을 가질 수 없다.
- 참조 무결성
 - 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다.
즉 릴레이션은 참조할 수 없는 외래키 값을 가질 수 없다.
 - 외래키와 참조하려는 테이블의 기본키는 도메인과 속성 개수가 같아야 한다.
- 도메인 무결성
 - 특정 속성의 값이, 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정.

답 : 데이터 무결성**PATH 16**

종류	정의
JMeter	HTTP, FTP, LDAP 등 다양한 프로토콜을 지원하는 안전성, 확장성, 부하, 기능 테스트 도구
LoadUI	HTTP, JDBC 등 주로 웹서비스를 대상으로 서버모니터링을 지원하는 UI를 강화한 부하 테스트 도구
OpenSTA	HTTP, HTTPS 지원하는 부하 테스트 및 생산품 모니터링 도구

답 : JMeter

PATH 17**1. System.out.println()**

- 해당 내용을 문자로 출력, 이후에 **개행(줄바꿈)**하는 출력문
- 문자의 경우 쌍따옴표(" ")로 지정하며, 변수 등의 경우에는 쌍따옴표 없이 직접 선언

ex) 문자열 변수 혼용 사용 시, 문자열 + 변수

```
System.out.println("값 : " + 10);
```

2. System.out.printf() or System.out.print()

- 일반 출력문(미개행)
- 출력하는 변수에 대해 서식 지정자를 통해 지정(=C언어)

```
ex> System.out.printf(" 값 : ");  
      System.out.printf("%d", 10);
```

출력 → 값 : 10

**답 : Hello
PATH**

PATH 18**1. TELNET**

- 네트워크 기반의 원격 통신 프로토콜
- 데이터 전송 시 **암호화 작업이 없어서, 보안성이 낮음**
- 포트번호 : 23

2. SSH(Secure Shell)

- 원격 통신에 이용되는 프로토콜
- 데이터 전송 시 **암호화 기법을 통해, 보안성이 높음**
- 포트번호 : 22

3. rlogin(Remote Login) ≈ TELNET

- UNIX 시스템 간의 같은 네트워크 안에서 사용하는 프로토콜
- 포트번호 : 513

답 : Telnet

PATH 19

1. SELECT 문의 문법순서와 실행 순서

SELECT	컬럼	GROUP BY	그룹화할 컬럼
FROM	테이블	HAVING	함수
WHERE	조건	ORDER BY	컬럼 정렬방법

- 문법 순서(작성순서)

SELECT → **FROM** → **WHERE** → GROUP BY →

HAVING → **ORDER BY**

- 실행 순서

FROM → WHERE → GROUP BY → HAVING →

SELECT → ORDER BY

답 : FROM, WHERE ORDER BY

PATH 20

종류	정의
Oracle	미국 오라클사에서 개발한 데이터베이스 관리시스템으로 유료이다. 리눅스/ 유닉스/윈도 모두를 지원하며 대형 시스템에서 많이 사용함
SQL Server	마이크로소프트사에서 개발한 관계형 데이터베이스 시스템이다. 마이크로소프트사 제품이기 때문에 윈도우 서버에서만 구동이 되며, 마이크로소프트사의 개발언어인 C# 등과 가장 잘 호환된다.
MySQL	썬 마이크로시스템에서 소유했던 관계형 데이터베이스 시스템이었으나 오라클에서 인수하였다. 리눅스, 유닉스, 윈도에서 모두 사용이 가능하고 오픈 소스 기반으로 개발되었다.
Maria DB	MySQL 출신 개발자가 만든 데이터베이스로 MySQL과 완벽히 호환된다.

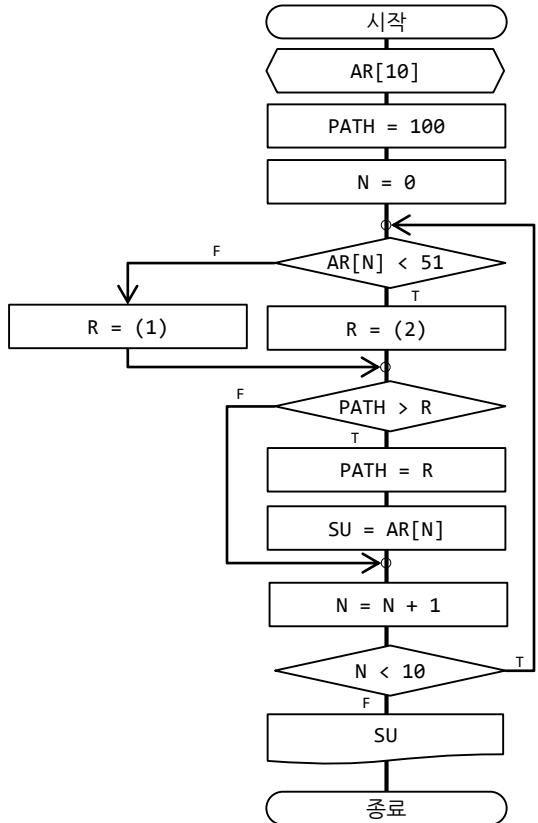
답 : RDBMS

PATH

정보처리기능사 실기
예상문제 11회

01. 알고리즘

배열 AR의 10개의 값 중 51에 가장 가까운 근사값을 갖는 배열을 찾아서 해당하는 배열의 값을 출력하는 알고리즘이다. 다음 빈칸에 들어갈 가장 알맞은 답안을 쓰시오.



02. 다음과 같은 <출력형태>가 나오도록 C언어로 프로그램을 작성할 때 빈칸에 들어갈 C언어 표현을 작성하시오.

<출력형태>

```

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
  
```

```

#include <stdio.h>
void main(){
    int i,n;
    for(n=1;n<= ( 1 ) ;n++){
        for(i=1;i<= ( 2 );i++){
            printf("%d ",i);
        }
        printf("\n");
    }
}
  
```

03. 다음 JAVA 소스코드를 분석하여 출력 결과를 작성하시오.

```
class Path{  
public static void main (String[] args){  
    int a = 2;  
    a += 5;  
    a -= 1;  
    a *= 2;  
    a /= 4;  
    a %= 3;  
    System.out.print(a); } }
```

04. 다음 C언어의 소스코드를 분석하여 i 변수에 3이 입력될 때
출력 결과를 작성하시오.

```
#include <stdio.h>  
int repath(int n){  
    int i;  
    if(n<1)  
        return 2;  
    else  
        i=(2*repath(n-1))+1;  
    printf("%d\n",i);  
    return i;  
}  
void main(){  
    int i;  
    scanf("%d",&i);  
    repath(i);  
}
```

05. 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include<stdio.h>
int main(){
    int a = 0, sum = 0;

    while( a < 10 ){
        a++;
        if(a%2 == 0)
            continue;
        sum += a;
    }

    printf("sum = %d", sum);

    return 0;
}
```

06. 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include<stdio.h>

void main(){
    int x=5; int y=5; int a;

    a = ++x + y--;
    printf("%d\n",a);
}
```

- 07.** 아래 보기의 <학생> 테이블에서 2학년들의 학과를 ‘컴퓨터과’로 갱신하는 SQL문의 빈칸을 완성하시오.

<학생>

학번	성명	학년	과목
200602	이연	3	토목과
201112	남지아	2	사회복지과
200419	이랑	2	호텔서비스과
200722	이무기	3	생명공학과
201225	기유리	1	비서과

```
UPDATE 학생 ( ① ) 학과 = '컴퓨터과' ( ② ) 학년 = 2;
```

- 08.** 다음은 학생들의 성적을 나타낸 표이다.
아래의 릴레이션을 보고, 기수(Cardinality)를 구하시오.

순번	이름	나이	종목
1	정준형	23	유도
2	송시호	24	무용
3	정재이	26	태권도
4	김대호	21	유도
5	정난희	21	역도

09. 다음 보기의 해당하는 용어를 쓰시오.

악의적인 사용자가 보안상의 취약점을 이용하여, 임의의 SQL 문, 사용자 인증을 우회하는 조건, 운영체제 명령을 삽입하고 실행되게 하여 데이터베이스가 비정상적인 동작을하도록 조작하는 행위

10. <보기>를 참고하여, 학생명단 테이블을 생성하는 SQL문을 완성하시오.

테이블은 학번, 이름, 학년, 점수로 구성

학번과 학년, 점수는 정수형 자료이며, 이름은 문자형 자료이다.

이름 속성은 공백이 있을 수 없다.

학년 속성의 기본값은 1이다.

학번을 기본 키로 지정한다.

```
CREATE TABLE 학생명단 (
    학번 INT, 점수 INT,
    이름 CHAR(10) ( ① ),
    학년 INT ( ② ) 1,
    ( ③ ) KEY(학번)
);
```

- 11.** 다음 <학생> 테이블에서 학년이 1학년 이거나 3학년인 학생의 학과를 검색하되 중복된 값을 제거 하는 구문이다.
빈칸에 들어갈 단어를 작성하시오.

```
SELECT ( ① ) 학과  
FROM 학생  
WHERE 학년 ( ② ) (1,2);
```

- 12.** 집합연산자 중 여러 개의 SQL문의 결과에 대한 합집합으로, 결과에서 모든 중복된 행은 하나의 행으로 만드는 연산자는 무엇인가?

13. 다음의 <보기>를 참고하여, 해당하는 용어를 작성하시오.

동영상 압축 포맷 중 하나인 MPEG에서 멀티미디어 콘텐츠를 표현하기 위한 표준이며 멀티미디어 자료를 효율적으로 저장하고 검색할 수 있도록 하는 것

14. 다음 <보기>에서 설명하는 프로토콜을 작성하시오.

인터넷 상에서 클라이언트가 원격지의 서버에 데이터를 업로드 및 다운로드 하기 위해 사용되는 프로토콜의 하나로서, 20번 포트와, 21번 포트를 사용하여 작동한다.

보통의 방식으로는 21번 포트를 통해 클라이언트가 서버에 접속하게 되면 제어와 관련된 세션을 사용하게 되며, 데이터를 전송하기 위해서 20번 포트를 사용하여 서버에서 클라이언트로의 접속을 하는 형태로 이루어진다.

15. 다음 보기를 읽고 해당하는 용어를 작성 하시오.

인터넷을 통해 언제 어디서나 방송/프로그램 등의
미디어 컨텐츠를 시청(소비)할 수 있는 사용자 중심적인 서비스

16. 다음 보기를 읽고 해당하는 용어를 작성 하시오.

예시를 통해 학습하는 것을 컴퓨터가 수행할 수 있도록 가르치는
기계 학습 알고리즘의 집합

17. 다음 보기의 내용을 읽고 해당하는 용어를 작성 하시오.

노트북, 태블릿, 스마트폰과 같은 장치를 텔레비전이나 모니터와 같은 디스플레이에 무선으로 접속하기 위한 표준

18. 다음 보기에서 설명하는 용어를 작성 하시오.

컴퓨터 네트워크에 연결된 컴퓨터 기억장치, 직접 연결하지 않고 네트워크를 통해 데이터를 주고 받는 저장장치이다.

19.

다음 설명 하는 용어를 작성하시오.

프로그램 실행 여부를 확인하는 정적 테스트 방법 중 하나이다.
소프트웨어 검토를 위해 미리 준비된 자료를 바탕으로, 정해진
절차에 따라 개발자가 모집한 전문가들이 소프트웨어를 검토하는
행위를 무엇이라 하는가?

20.

다음 보기에서 설명하는 소프트웨어 테스트의 기본 원칙을
작성하시오.

동일한 테스트 케이스로 반복 실행하면 결함을 발견할 수 없으므로 주기적으로 테스트 케이스를 리뷰하고 개선해야 한다.

1.	(1) AR[N]-51 (2) 51 – AR[N]
2.	(1) 5 (2) n
3.	0
4.	5 11 23
5.	25
6.	11
7.	① : SET ② : WHERE
8.	5
9.	SQL Injection / SQL 삽입 공격
10.	① : NOT NULL ② : DEFAULT ③ : PRIMARY

11.	① : DISTINCT ② : IN
12.	UNION
13.	MPEG-7
14.	FTP
15.	OTT(Over The Top)
16.	딥 러닝(Deep Learning / 심층학습)
17.	미라캐스트(Miracast)
18.	NAS (Network Attached Storage)
19.	워크스루 / 겸토회의
20.	살충제 패러독스 (Pesticide Paradox)

PATH 01**1. 사이값 / 근사값**

- 비교할 숫자가 내 배열원소보다 큰 경우 비교할 숫자에서 내 배열 수를 뺀다.
(ex 비교숫자 51 / 배열원소 A[1] = 20 --> 사이 값 = 51 - A[1])
- 비교할 숫자가 내 배열원소보다 작은 경우 내 배열 수에서 비교할 숫자를 뺀다.
(ex 비교숫자 51 / 배열원소 A[2] = 80 --> 사이 값 = A[2] - 51)

예시 코드

```
#include <stdio.h>
void main()
{
    int ar[10]={77,33,22,11,55,99,66,44,88};
    int path=100,n=0,su,r;
    while(n < 10) {
        if(ar[n] < 51) { r = 51 - ar[n]; }
        else { r = ar[n]-51; }
        if(r < path) {
            path = r;
            su = ar[n];
        }
        n = n + 1;
    }
    printf("%d",su);
}
```

답
AR[N] – 51
51 – AR[N]

PATH 02**1. 이중 FOR문**

- 이중 FOR문 코드식에서는 주로 첫 번째 FOR문이 행(줄)을 두 번째 FOR문은 열(칸)을 담당합니다.
- 코드상 줄에 맞게 안에 변수가 종료되어야 해서 첫 번째 FOR문의 변수 n을 입력해 줍니다

코드 분석

```
for(n=1; n<=5 ;n++) \\ 전체 줄 수 (5)

for(i=1;i<=n;i++) \\ 줄 수에 맞게 숫자 증가 1부터 줄 수 까지
\\ 1줄 1개 / 2줄 2개 ...

printf("\n"); \\ 줄 바꿈
```

답 : (1) 5 (2) n

PATH 03**1. 산술 연산자**

- 특정 기호를 이용하여 사칙연산을 하는 작업을 의미합니다.
- +[덧셈], -[뺄셈], *[곱셈], /[나눗셈], %[나머지] 등이 있습니다.

2. 복합 대입 연산자

- 산술 연산자와 대입 연산자를 간결하게 사용하는 작업입니다.

ex) 'A = A+1'를 'A += 1'로 표현

3. 코드 분석

```
int a = 2;      // num : 2
a+=1;          // a = a + 5  →  7
a-=2;          // a = a - 1  →  6
a*=3;          // a = a * 2  →  12
a/=4;          // a = a / 4  →  3
a%=5;          // a = a % 3  →  0
System.out.print(a);    // a = 0
```

답: 0

PATH 04**1. 재귀 함수**

- 하나의 Method(함수)에서 **자기 자신을 다시 호출**하여 해당 구문을 다시 수행하는 함수입니다.

```
if(n<1) // n 이 1보다 작아지면 2를 반환
    return 2;
else
    i=(2*repath(n-1))+ 1;
    ↳ (2*repath(3-1))+1 ③ 11값은 받아서 연산 -> 23
    ↳ (2*repath(2-1))+1 ② 5값을 받아서 연산 -> 11
    ↳ (2*repath(1-1))+1 ① n값이 1보다 작아 2를 반환 -> 5
※ 순서는 ①->②->③로 출력 됨
```

답: 5

11

23

PATH 05

1. 반복문 제어

- 반복문에서 반복된 흐름을 벗어나기 위해 사용합니다.
- 대표적으로 **break**와 **continue**문이 있습니다.
 - * **break** : 해당 반복문을 멈추고, **탈출**하는데 사용합니다.
 - * **continue** : 반복문을 유지하면서, 현재 흐름을 **건너뛰**니다.

2. 코드 분석

```
while( a < 10 ){      // 반복 조건 : a가 10 이하
    a++;                // a + 1
    if(a%2 == 0)        // a가 짝수(2로 나눈 나머지가 0)일 때
        continue;         // 코드 건너뛰기 : while문 복귀
    sum += a;             // sum = sum + a
}
```

답: 25

PATH 06

1. 증감 연산자

증감 연산자는 피연산자를 ++로 1씩 증가시키며, --로 1씩 감소시킵니다. 또한, 결합 위치에 따라 전위 연산자와 후위 연산자로 나뉩니다.

- 전위 연산자(++a / --a) : 피연산자 증감 후, 연산 수행
- 후위 연산자(a++ / a--) : 연산 수행 후, 피연산자 증감

```
int x=5; int y=5; int a;      // x:5  y:5
a = ++x + y--;               // a = 6 + 5
printf("%d\n",a);            // a : 11
```

답: 11

PATH 07**1. UPDATE**

- 데이터베이스 조작어인 DML의 한 종류로, 데이터를 변경할 때 사용하는 명령어입니다.

2. DML

- 검색

```
SELECT column_name FROM table_name
```

- 삭제

```
DELETE column_name FROM table_name
```

- 갱신

```
UPDATE table_name
```

```
SET column1 = value1, column2 = value2...
```

- 추가

```
INSERT INTO table_name(col1,col2...)  
VALUE (value1,value2...)
```

답 : ① SET ② WHERE

PATH 08**1. 속성(Attribute)**

Red	Red	Red	Red

2. 튜플(Tuple)

Red			

릴레이션의 속성(Attribute)의 개수를 차수(Degree)라 하며,

튜플(Tuple)의 개수를 카디널리티(Cardinality:기수)라 부릅니다.

TIP) 속성은 컬럼 혹은 필드로도 표현합니다.

튜플은 행 혹은 레코드로 불리기도 합니다.

※ 본 문제의 속성은 4개이므로 차수는 4입니다.

첫 행을 제외한 튜플의 수는 총 5줄이므로 카디널리티는 5입니다.

답 : 5

PATH 09**1. SQL Injection**

- 응용 프로그램 보안 상의 허점을 의도적으로 이용해, 악의적인 SQL문을 실행되게 함으로써 데이터베이스를 비정상적으로 조작하는 가장 대표적인 방법

2. SQL Injection 유형

- Error based : 논리적 에러를 이용한 인젝션
- Union based : UNION 명령어를 이용한 인젝션
- Boolean based : Blind 라고 하며 공격자에게 보이지 않게 사용하는 인젝션

답 : SQL Injection / SQL 삽입 공격

PATH 10**SQL 분석**

```
CREATE TABLE 학생명단 ( // 학생명단 테이블을 생성한다.  
학번 INT, 점수 INT, // 학번과 점수는 정수형이다.  
이름 CHAR(10) NOT NULL, // 이름은 문자형이며  
                           공백이 있을 수 없다.  
학년 INT DEFAULT 1, // 학년은 정수 형으로 기본값이 1 이다.  
PRIMARY KEY(학번) // 학번은 기본 키로 한다.  
);
```

**답 : ① NOT NULL
② DEFAULT
③ PRIMARY**

PATH 11**1. IN (A , B)**

- SQL 구문의 조건 문에서 지정한 값 중 하나라도 일치하는 것을 반환합니다.

ex) 학년 IN(1,2); → 학년이 1학년 이거나 2학년

2. BETWEEN A AND B

- SQL 구문의 조건 문에서 지정한 두 수 사이의 숫자를 반환합니다.

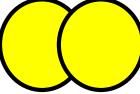
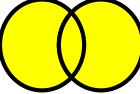
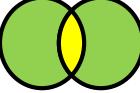
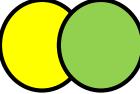
ex) 나이 BETWEEN 10 AND 50 → 10세부터 50세까지 사이

3. DISTINCT

- 데이터의 중복을 제거

답 : ① DISTINCT ② IN

PATH 12

집합연산자	형태	정의
UNION (합집합)		중복 행이 제거된 집합
UNION ALL (합집합)		중복 행이 제거되지 않은 집합
INTERSECTIO N (교집합)		두 쿼리 결과에 공통적으로 존재하는 집합
MINUS (차집합)		첫 쿼리에 있고 두 번째 쿼리에 없는 집합

답 : UNION

PATH 13**1. MPEG(Moving Picture Experts Group)**

- MPEG은 ISO 및 IEC 산하에서 비디오와 오디오 등 멀티미디어의 표준의 개발을 담당하는 소규모의 그룹

종류	정의
MPEG-1	최초의 비디오와 오디오 표준, CD의 표준으로 쓰임
MPEG-2	TV 방송을 위한 표준, 디지털 방송, DVD 비디오 등
MPEG-4	MPEG-2 를 확장하여 3D 콘텐츠 등을 포함하며, H.264라고도 불림
MPEG-7	멀티미디어 콘텐츠를 기술하기 위한 프로그램 동영상 검색이나 전자 상거래 등에 적합하도록 개발된 차세대 동영상 압축 재생기술이다.
MPEG-21	멀티미디어 애플리케이션 용 프레임워크라고도 표현한다. MPEG 관련 기술을 통합하여 디지털 콘텐츠의 제작 및 유통, 보안 등의 모든 과정을 관리할 수 있게 하는 기술

답 : MPEG-7

PATH 14**1. 대표적인 포트 번호**

포트	프로토콜
20	FTP(File Transfer Protocol) 데이터 포트
21	FTP(File Transfer Protocol) 제어 포트
22	SSH(Secure Shell) - 암호화 된 원격 통신
23	Telnet - 암호화 되지 않은 원격 통신
25	SMTP - 이메일 송신 사용
53	DNS(Domain Name System)
80	HTTP(HyperText Transfer Protocol) - 웹 페이지 전송
110	POP3 - 이메일 수신 사용

답 : FTP

PATH 15**1. Over The Top(OTT)**

- 인터넷을 통해 시간과 장소의 구애 받지 않고 시청할 수 있는 서비스
 - 하나의 플랫폼에만 종속되지 않고 PC, 스마트폰, 태블릿, 콘솔 게임기 등 다양한 플랫폼을 지원한다는 의미도 포함되어 있음
- ex) 넷플릭스, Apple TV, 디즈니, 왓챠, wavve 등

답 : OTT (Over The Top)

PATH 16**1. 딥 러닝(Deep Learning)**

- 인공 지능의 머신 러닝 기법
- 여러 비선형 변환기법의 조합을 통해 높은 수준의 추상화
- 다량의 데이터나 복잡한 자료들 속에서 핵심적인 내용 또는 기능을 요약하는 작업을 시도하는 기계 학습 알고리즘의 집합
- 사람의 사고방식을 컴퓨터에게 가르치는 기계학습의 한 분야

답 : 딥 러닝(Deep learning) / 심층학습

PATH 17**1. 미라캐스트(Miracast)**

- 무선으로 디스플레이를 연결하는 기술(미러링의 일종)
- 휴대폰이나 태블릿등 장치의 화면 영상과 소리를 다른 디스플레이에 연결하는 **기술**

2. 크롬캐스트(Chromecast)

- 구글에서 만든 멀티미디어 스트리밍 어댑터
- 디스플레이 장치의 직접 연결하여 스트리밍 또는 재생을 하게 하는 **장치**

답 : 미라캐스트(Miracast)

PATH 18**1. NAS(네트워크 결합 스토리지)**

- 네트워크로 연결하는 하드디스크 이자 **저장장치**
- 여러대의 기기의 데이터를 저장하고 공유하며 원격으로 실행 가능
- 반대 개념은 직접 연결해서 쓰는 DAS(Direct Attached Storage)

2. 클라우드 컴퓨팅

- 인터넷을 이용해 가상화된 컴퓨터의 시스템리소스를 요구하여 즉시 제공 받는 것
- 인터넷 기반 컴퓨팅의 일종으로 자신의 컴퓨터가 아닌 인터넷에 연결된 다른 컴퓨터로 처리하는 **기술**

답 : NAS(Network Attached Storage)

PATH 19**1. 워크스루 와 인스펙션 비교**

구분	워크스루	인스펙션
목적	산출물 평가 및 개선	결함 파악 및 제거
수행 조건	팀이나 관리자의 필요 시	완성도 기준 이상 시
결합 수정 여부	저자 결정	모든 결합 제거
검토자 인원	2 - 7 명	3 - 6 명
참여자	기술 전문가 및 동료	동료
검토 인도자	저자	진행자
검토 준비 여부	준비하지 않음	체크리스트 활용
검토 속도	빠름	느림
발표자	저자	리더
지표 수집 여부	수집하지 않음	수집함
보고서	워크스루 보고서	결함 리스트
데이터 측정 여부	권장	필수
체크리스트 사용 여부	사용하지 않음	사용함

답 : 워크스루 / 검토회의**PATH 20****1. 살충제 패러독스 (Pesticide Paradox)**

- 지속적인 살충제 살포로 내성이 생긴 벌레가 죽지 않는 현상
- 애플리케이션 테스트에서도 동일한 테스트에 대한 비정상적인 결합 검수가 이루어질 수 있습니다. 이러한 현상을 방지하기 위해서는 정기적인 리뷰와 개선이 필요하고, 다양한 시각과 방법으로 테스트하는 것이 필요합니다.

2. 테스트의 7가지 원칙

- 1) 테스트는 결함이 존재함을 밝히는 활동이다.
- 2) 완전한 테스트는 불가능
- 3) 초기 테스트
- 4) 결함 집중
- 5) **살충제 패러독스**
- 6) 테스트는 정황의 의존적이다.
- 7) 오류-부재의 궤변

답 : 살충제 패러독스 (Pesticide Paradox)

PATH

정보처리기능사 실기
예상문제 12회

01.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

02.

HTML의 단점을 보완하여, 애니메이션 강화 및 사용자와의 상호 작용에 초점을 두었으며, 동적인 웹 페이지 조정이 가능한 언어는 무엇인가?

```
#include <stdio.h>

int main(void) {
    int arr[5] = {0, };
    int result;

    result = sizeof(arr)/sizeof(int);

    printf("%d", result);

    return 0;
}
```

03. 다음 JAVA 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class function
{ public static void main (String[] args)
{
    char str = 'B';
    switch(str)
    {
        case 'A':
        System.out.println("1번"); break;
        case 'B':
        System.out.println("2번"); break;
        default:
        System.out.println("누락");
    }
}
```

04. 다음 보기지를 읽고 해당하는 용어를 작성 하시오.

컴퓨터와 같은 기기의 구성 요소를 컴퓨터가 동작하는 도중에 제거하고 바꾸는 기능이다. 대표적으로 항상 전원을 유지해야 하는 서버 컴퓨터나 RAID 시스템 등이 있다.

05. OSI 7계층의 기능을 모두 구현 가능한 장치를 뜻하며, 통신의 가장 처음과 마지막으로써의 출발과 목적지 장치를 무엇이라 하는가?

06. 다음 C언어에서 각 입력문마다 Hello World를 입력하여 실행되는 결과를 입력하시오.

```
#include <stdio.h>

int main(void) {
    char str1[20];
    char str2[20];

    scanf("%s\n", str1);
    fgets(str2, 12, stdin);

    printf("%s\n", str1);
    printf("%s\n", str2);
    return 0;
}
```

07.

다음이 설명하는 단어를 적으시오.

현대 사회에 쉽게 발병되는 공포증 중 하나로 휴대폰을 가지고 있지 않을 때 알 수 없는 불안감과 공포감에 휩싸이게 된다고 한다.

08.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>
#include <string.h>

int main(void) {
    int *arr = "ABCDEFG";

    printf("%d", strlen(arr));
    return 0;
}
```

09. 다음은 SQL문에 사용되는 ORDER BY절의 대한 설명으로
빈칸에 알맞은 단어를 적으시오.

ORDER BY의 경우 테이블을 내림차순 혹은 오름차순으로 정렬해야 할 때 사용됩니다. 이때 ORDER BY와 정렬하고자 하는 열의 이름 뒤에 원하는 차순을 적어주게 되며 아무것도 적지 않았을 경우, () 명령어가 기본값이 됩니다.

10. 다음 JAVA 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class function
{
    public static void main (String[] args)
    {
        System.out.println(5 % (4 + 2) * 3);
    }
}
```

11. 다음이 설명하는 데이터베이스 용어는 무엇인가?

데이터베이스 내에 데이터들이 불필요하게 중복되어 예기치 못한 이상(Anomaly)현상을 방지하기 위해 (①)를 거친다.

12. 다음에서 설명하는 용어를 쓰시오.

후보키 중에서 특별히 선정된 키인 기본키가 선택되고 남은 후보키들

13.

다음에서 설명하는 용어를 쓰시오.

조인은 크게 2가지로 구분 되는데, (①)은 일반적으로 교집합 형태인 내부 조인과 다른 모양을 하고 있으며, 조건에 만족하지 않더라도 결과로 출력하기 위해 사용하는 조인이다.

14.

다음은 학생 테이블에서 A로 시작하는 문자의 속성을 전부 가져오는 데이터베이스 구문이다.
쿼리의 빈칸을 완성하시오.

순번	점수	순위
1	5.4	C1
2	3.2	S1
1	4.2	E3
1	2.2	S2
3	1.8	C3

SELECT () FROM 학생 WHERE 순위 LIKE 'C%'

15. 다음은 DCL 에 관한 설명이다.
빈칸을 완성하시오.

1. (①)은 실행된 명령의 수행 결과를 물리적으로 존재하는 디스크에 저장한 뒤, 작업이 완료됐음을 알린다.
2. (②)은 데이터베이스 사용자의 사용 권한을 박탈(취소)한다.
3. (③)은 데이터베이스의 조작 작업의 종료가 비정상적임이 감지되었을 때 원래 상태로 복구시킨다.
4. (④)은 데이터베이스 사용자의 사용 권한을 부여한다.

16. 다음 요청에 따라, 데이터베이스 질의를 완성하시오.

성적 뷰(View)를 제거하고 연관된 뷰들도 연쇄적으로 제거하시오.

(①) VIEW 성적 (②)

**17.**

다음 <보기>에서 설명하는 용어를 작성하시오.

18.

블로그와 뉴스 같은 콘텐츠를 사이트에 직접 방문할 필요없이 한 곳에 모아 볼 수 있게 해주는 서비스를 무엇이라 하는가?

토플로지중의 하나로서, 중앙제어장치에 다른 장치들이 연결되어 있는 모습이나, 대부분의 장치는 중앙 허브에 연결되어 있는 2차 허브와 연결되어 있는 형태를 이루고 있는 접속 형태이다.
해당 구성의 경우 장점으로는 중앙장치에 더 많은 장비를 연결할 수 있으며, 제어가 간편하며, 관리나 네트워크 확장이 쉬우나,
중앙에 트래픽이 집중되게 되면, 병목 현상의 발생이 쉬워진다.

19. USB 2.0 포트는 검은색 혹은 흰색을 띠고 있다. 이 다음로 개선된 전자기기 통신 USB 표준 3번째의 USB 3.0은 무슨 색을 띠고 있는가?

20. 다음 JAVA 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
class Function{
    public static void main (String[] args)
    {
        System.out.println(4 << 1);
        System.out.println(4 << 2);
    }
}
```

1.	5
2.	동적 HTML(DHTML)
3.	2번
4.	핫스왑(Hot Swap)
5.	종단장치 (End System or End Station)
6.	Hello Hello World
7.	노모포비아(Nomophobia)
8.	7
9.	ASC
10.	15

11.	정규화(Normalization)
12.	대체키
13.	Outer Join
14.	*
15.	① : COMMIT ② : REVOKE ③ : ROLLBACK ④ : GRANT
16.	① : DROP ② : CASCADE
17.	트리거
18.	RSS
19.	파란색
20.	8 16

PATH 01**1. sizeof 함수**

- 대상의 크기를 Byte(8 bit) 단위로 구하는 함수입니다.

sizeof(자료형)	sizeof(변수) or sizeof 변수
<pre>1 printf("%d", sizeof(int)); 2 // 4 출력 int는 4byte의 크기를 가짐 하단의 표 참조</pre>	<pre>1 int num[3]; 2 printf("%d", sizeof(num)); // 12 출력 4(int) * 3(배열의 갯수) int가 num에 총 3개 있습니다.</pre>

2. 대표적인 자료형의 크기

종류	역할
정수형	char (1byte) : -128 ~ 127까지 저장 short (2byte) : -32768 ~ 32767까지 저장 int (4byte) : -2,147,483,648 ~ 2,147,483,647
실수형	float (4byte) : $3.4 \times 10^{-37} \sim 3.4 \times 10^{38}$ double (8byte) : $1.7 \times 10^{-307} \sim 3.4 \times 10^{308}$
문자형	char (1byte)

답 : 5**PATH 02****1. HTML**

- 대표적 마크업 언어로서, 하이퍼텍스트 문서를 만들기 위해 사용

2. 동적HTML(DHTML)

- 사용자의 편의를 위해 드래그 앤 드롭(Drag & Drop) 기능 등을 지원함
- 사용자의 특성에 따른 동적 웹페이지 설정 가능

3. HTML5

- 이전 버전들의 단점을 보완한 마크업 언어로서, 불필요한 기능을 삭제하였고, 기존의 코딩 방식을 간결하고 명확하게 변경
- 이전 버전인 XHTML에 있던 XML 문법을 모두 삭제

답 : 동적 HTML(DHTML)

PATH 03**1. switch case문**

- switch(변수)의 값을 확인하여, case의 값과 일치하게 되면 해당 실행문을 인식
- case의 값은 줄맞음을 할 때 :(콜론)을 통해 구분하게 되며, 다른 구문과 달리 중괄호({},) 없어서 라인 구분을 할 수 없기에, break를 이용하여 해당 구문을 구분합니다.

※ break 구문이 없으면, 멈추지 않고 다음 구문을 출력

구문
<pre>char str = 'A'; switch(str) // 변수 지정 { case 'A': // 해당 값과 일치 →System.out.printf("1번"); // 해당 라인 실행 case 'B'; //break가 없어서 조건식을 읽지않고 다음 구문까지 실행 →System.out.printf("2번"); // 1번2번 출력 }</pre>

답 : 2번

PATH 04**1. 핫스왑(Hot Swap)**

- 시스템 전원이 켜진 상태에서 안전하게 장치를 제거가 가능하며, 새로운 장치를 연결할 수 있는 기능입니다.

ex) 핫스왑 베이 등을 이용하여 HDD(하드디스크), SSD 등 교체

1. 핫플러그(Hot Plug)

- 시스템 전원이 꺼진 상태에서 제거 및 새로운 장치를 연결해야하며, 만약 켜진 상태에서 장치를 제거하면 데이터 손실 우려가 있습니다.

ex) USB

답 : 핫스왑(Hot Swap)

PATH 05

1. 터미널 or 단말기

- 디스플레이에 데이터 등의 입출력 등의 기능을 수행하는 장치

2. 종단장치(End System)

- 네트워크의 가장 처음과 마지막으로써의 종단에 해당하는 장치
- OSI 7계층의 기능을 모두 구현 가능
- ex) PC, PDA, 휴대폰

3. 중간장치

- OSI 7계층의 기능 중 하위 3계층만의 기능을 수행하는 장치

답: 종단장치(End System or End Station)

PATH 06

1. 입력 (Scanf)

- 입력 내용에 여러가지의 형식을 지정할 수 있음.
- **scanf는 입력을 할 시에, 공백을 문자로 인식하지 않고 끝냄**

구문

`scanf("서식지정자", 변수의 주소)`

ex) `int a;`
`scanf("%d" , &a);` ※ &란 주소 연산자로서 메모리 주소를 가져옴

2. 입력 (fgets)

- 함수의 크기에 제한을 지정하여 사용하는 문자를 입력하는 구문
- 입력 시, 줄바꿈(Enter)를 입력할 때까지 해당 한 줄을 모두 읽는다.

구문

`fgets(변수, 함수의 크기 , 텍스트 포인터)`

ex) `int a[20];`
`fgets(a, 12, stdin);` ※ stdin이란 표준 입력장치[키보드]를 의미함

**답: Hello
Hello World**

PATH 07

1. 노모포비아(Nomophobia)

- No, Mobile, Phobia의 세 단어를 합성한 신조어로, 쉽게 말해 휴대폰을 가지고 있지 않을 때 공포감을 느낀다는 뜻이다. 휴대폰으로 인한 짧은 자극에 익숙해져 있기 때문에 장기간 집중하는 능력이 떨어져 학업성적이나 업무성과에 큰 영향을 미치곤 한다.

답 : 노모포비아(Nomophobia)

PATH 08

1. 헤더(header)파일

- 헤더(header)파일이란 특정 기능을 가진 함수의 모음.

헤더 파일	자주 사용하는 함수
#include<string.h> (문자열 관련)	strlen(변수) // 문자열의 길이 strcpy(변수, 복사할 값) // 복사할 값을 변수에 저장 strcmp(변수, 비교할 값) // 변수를 기준으로 값을 비교함 변수<값 : 음수 변수>값 : 양수 변수==값 : 0 strtok(변수, 구분할 기준) // 기준으로 변수의 값을 자름 ex) strtok("hello World", " "); 출력 : hello strcat(변수, 이어붙일 값) // 변수의 값 뒤에 이어붙일 값을 이어줌.

답 : 7

PATH 09

1. ORDER BY

- ASC : 오름차순
- DESC : 내림차순

2. GROUP BY

- 테이블을 그룹화할 때 사용하는 구문이다. 그룹과 동시에 조건을 적용하고 싶을 경우 WHERE문이 아닌 HAVING문을 사용한다.

답 : ASC

PATH 10

1. 연산자 우선순위

우선 순위	연산자
1	증감연산자(후위), ()[괄호]
2	증감연산자(전위), ~[보수], ![논리], &[주소]
3	*[곱셈], /[나눗셈], %[나머지]
4	+[덧셈], -[빼셈]
5	<<, >>[시프트연산자]
6	>, >=, <, <=, ==, !![비교연산자]
7	&[비트 AND 연산자]
8	^[비트 XOR 연산자]
9	[비트 OR 연산자]
10	&&[논리 AND연산자]
11	[논리 OR 연산자]

답 : 15

PATH 11**1. 데이터베이스 정규화**

- 불필요한 데이터(Data Redundancy)를 제거하는 과정입니다.
- 논리적으로 데이터 저장합니다.

2. 정규화 과정**비정규 릴레이션**

1. 도메인을 분해
2. 부분적 함수 종속 제거
3. 이행적 함수 종속 제거
4. 결정자 이면서 후보키가 아닌 것 제거
5. 함수 종속이 아닌 디치 종속 제거
6. 후보키를 통하지 않은 조인 종속성 제거

1NF

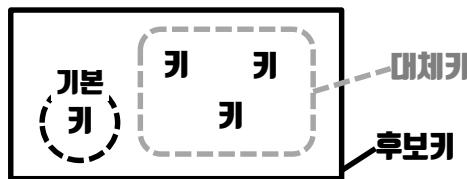
2NF

3NF

BCNF

4NF

5NF

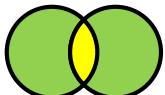
답 : 정규화(Normalization)**PATH 12****키(Key)**

1. 후보키(Key) : 유일성 / 최소성 만족
2. 기본키 : 후보키 중 선정된 키로 중복 값 입력이 불가능하다.
Null 값을 가질 수 없다.
3. 슈퍼키 : 유일성 만족 / 최소성 불만족
4. 대체키 : 기본 키로 선택되지 못한 후보 키
5. 외래키 : 다른 테이블의 행을 식별하는 키

답 : 대체키

PATH 13**1. INNER JOIN(내부 조인)**

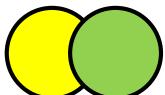
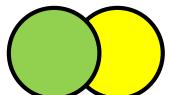
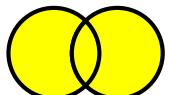
- 두 테이블에서 조인된 필드가 일치하는 행만을 표시



INNER JOIN

2. OUTER JOIN(외부조인)

- 조인 조건에 만족하지 않는 튜플도 결과로 출력하기 위한 조인

LEFT
OUTER JOINRIGHT
OUTER JOINFULL
OUTER JOIN**답 : OUTER JOIN(외부 조인)****PATH 14****1. 와일드 카드**

- Where 문에서의 와일드 카드는 %는 “모든 문자”를 의미하며, _는 “단일 문자”를 의미, Select 문의 뒤에 들어갈 경우에는 *로 전체를 의미하는 와일드카드를 사용할 수 있다.

답 : *

PATH

PATH 15

데이터베이스 언어인 SQL(Structured Query Language)는 정의어(DDL), 조작어(DML), 제어어(DCL)로 구성되어 있습니다.

제어어 DCL(Database Control Language)는 DBMS 내에서 데이터의 보안, 무결성 등을 제어합니다.

대표적인 명령으로 COMMIT(작업완료), ROLLBACK(작업취소), GRANT(권한부여), REVOKE(권한해제)가 있습니다.

답: ① COMMIT ② REVOKE
 ③ ROLLBACK ④ GRANT

PATH 16

1. **DROP** : 테이블 및 뷰 등의 삭제

- 테이블 및 뷰 등의 내부 개체 수의 상관없이 삭제합니다.

2. **CASCADE** : 종속

- 개체를 변경 또는 삭제할 때 대상 개체가 다른 개체를 참조하고 있을 경우 연쇄적으로 함께 변경 또는 삭제합니다.

답 : ① **DROP** ② **CASCADE**

PATH 17**1. 성형**

- 네트워크 중앙에 허브가 위치하며, 다른 모든 노드들을 연결
- 중앙집중적인 구조이므로 유지 보수 및 확장에 용이

2. 버스형

- 모든 네트워크 노드 및 주변 장치들이 파이프와 같은 일자형 케이블에 연결되어 있는 구조

3. 트리형

- 성형의 변형으로 중앙제어 장치에 장치들이 연결되어 있으며, 2차 허브들을 통해 또 다른 장치들이 연결되어 있는 구조
- 특징으로는 성형과 비슷하게, 네트워크 확장이 용이하며, 중앙에 데이터가 많이 집중되면 트래픽이 발생한다.

4. 링형

- 모든 컴퓨터들이 하나의 링과 같이 연결되어 있는 구조

5. 그물형

- 중앙의 제어하는 노드가 없이 모든 노드들이 상호간 연결되어 있는 그물 모양의 구조

답 : 트리형**PATH 18****1. RSS(Rich Site Summary)**

- 이전에는 어떠한 사이트에서 정보를 얻기 위해서는 사이트에 방문하여 직접 얻어야 했지만, RSS 시스템이 적용된 프로그램이나 서비스를 이용하면 자동으로 정보를 수집해 주기 때문에 사용자는 사이트를 하나하나 방문할 필요없이 최신 정보를 한 자리에서 수집할 수 있다.

답 : RSS

PATH 19

1. USB 3.0

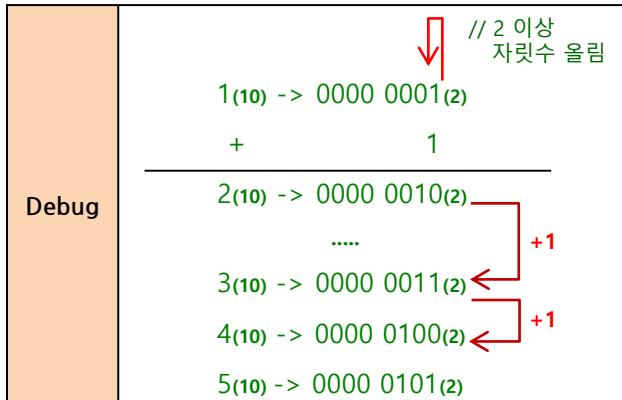
- 이전에 개발된 표준 USB 2.0의 속도와 약 10배의 차이를 보인다.
제조업체들은 USB 3.0과 2.0 단자를 구별하기 위해 SS라는 이니셜
과 파란색 코드로 구별할 것을 권고한다.

답 : 파란색

PATH 20

1. 이진수 변환 방법

- 숫자2를 모르는 덧셈 방법이라고 생각하시면 편합니다.



2. 자릿수 의미

128	64	32	16	8	4	2	1
0	0	0	0	0	1	0	1

↑ ↑ 5₍₁₀₎

이진수에서 1은 True라는 표현으로,
해당 자릿수가 1이면 해당 숫자를 더하시면 됩니다.

풀이과정

```
class Function{
    public static void main (String[] args)
    {
        System.out.println(4 << 1);
        4(2)_0000 0100(2)에서 자릿수를 좌측으로 한 칸 이동
        0000 1000(2) == 8 출력
        System.out.println(4 << 2);
        0000 0100(2)에서 자릿수를 좌측으로 두 칸 이동
        0001 0000(2) == 16 출력
    } // JAVA에서 println은 수행 후, 줄바꿈 기능을 수행
}
```

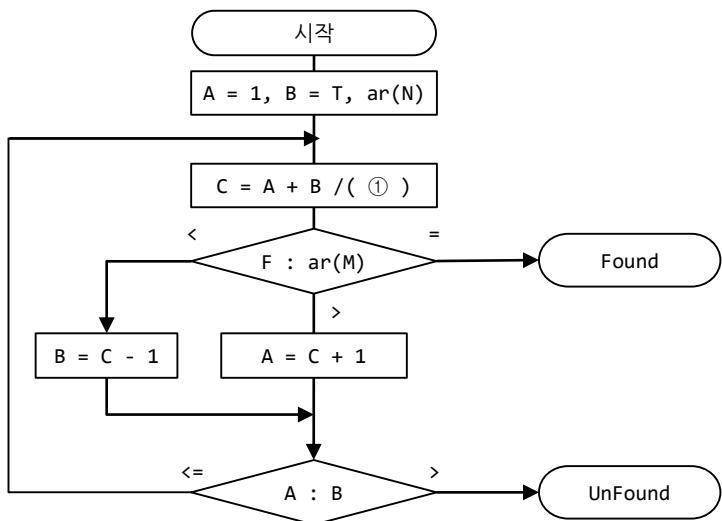
답 : 8
16

PATH

**정보처리기능사 실기
예상문제 13회**

- 01.** 오름차순 정렬된 데이터를 좌우로 이등분 후 목적하는 값의 범위를 점차 좁혀가며 검색하는 것을 이분 검색(Binary Search)이라고 한다. 다음의 알고리즘을 완성시키시오.

* 단, F는 찾고자 하는 키의 값이며 T는 전체 레코드의 수이다



- 02.** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```

#include <stdio.h>

int main(void) {
int ar[10], i=14, j=0;

while(i>0) {
ar[j] = i%2;
i = i/2;
j++;
}

while(j>0) {
j--;
printf("%d ",ar[j]);
}
}
  
```

03.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int num = 0b1111;
    printf("%d\n", num);
    printf("%o\n", num);
    printf("%x\n", num);
}
```

04.

다음은 SQL문의 ALTER TABLE 컬럼 명령어이다. 출석 테이블 내의 전화번호 컬럼을 삭제하려는 목적이라고 할 때, 빈 칸 안에 들어갈 명령어를 작성하시오.

ALTER TABLE 출석 (①) 전화번호

05.

빈 칸에 들어갈 말을 작성하시오.

- (①)은 마이크로소프트(Microsoft)가 개발한 관계형 데이터 베이스(RDBMS)이다.
- (①)은 Window Server에서만 구동되며 C#과 가장 높은 호환성을 보인다.

06.

다음은 사원 테이블과 직원 테이블을 이용한 JOIN문이다.
빈칸에 들어갈 단어를 작성하시오.

```
SELECT * FROM 사원 INNER JOIN 직원  
( ① ) 사원.ID = 직원.ID WHERE 사원.ID = "KKA"
```

07. 다음은 DB 트랜잭션의 특성인 ACID에 대한 설명이다.
빈칸에 들어갈 알맞은 단어를 작성하시오.

원자성(Atomicity) : 실행 중 중단되지 않게 하는 능력

일관성(Consistency) : 실행의 완료 후 데이터베이스 상태를 유지

(①)(Isolation) : 한 트랜잭션의 실행 중 다른 트랜잭션이
끼어들지 못하게 보장

지속성(Durability) : 실행에 성공한 트랜잭션은 영원히 반영

08. 다음 지문에서 설명하는 단어를 작성하시오.

개발과 운영 등의 역할로, 기획안 분석과 데이터베이스 관계모델의 설계 등을 도맡으며 테이블 생성 및 프로시저 작성 및 관리, 데이터 검색, 삽입, 삭제 등을 수행하는 사람을 지칭하는 용어는?

09. SQL문의 일종으로 데이터베이스를 정의하는 언어이며, 데이터베이스의 생성, 수정, 삭제 등의 전체 골격을 결정하는 명령어는 무엇인가?

10. 다음에서 설명하는 용어를 쓰시오.

- 참조하는 테이블에서 1개의 키에 해당하고, 참조하는 쪽의 관계 변수는 참조되는 쪽의 테이블 키를 가리킨다.
- 하나의 테이블에 포함될 수 있으며, 각각 다른 테이블을 참조할 수 있다.

- 11.** 다음은 학교 테이블에서 과목이 과학인 반을 A반으로 변경하는 SQL문을 완성하시오.

```
UPDATE 학교
SET 반 = 'A반'
( ① ) 과목 = '과학'
```

- 12.** 소프트웨어 내부 구조나 작동 원리를 모르는 상태에서 동작을 검사하는 방식으로, 올바른 입력과 올바르지 않은 입력을 통해 올바른 출력이 나오는지를 검사하는 단어는 무엇인가?

13.

다음 테스트 기법에 대한 빈칸에 알맞은 방식을 작성하시오.

구분	스텝	드라이버
시기	상위 모듈은 있지만 하위 모듈이 없을 때, 하위 모듈로의 테스트를 진행	상위 모듈 없이 하위 모듈만 있을 때 상위 모듈로의 테스트를 진행
방식	①	②
특징	작성이 쉬운 시험용 모듈	존재 하지 않는 상위 모듈간의 인터페이스 역할

14.

테스트 결과가 올바른지 판단하기 위해 사전에 정의된 참 값 을 대입하여 비교하는 기법 및 활동을 테스트 오라클 이라고 한다. 다음이 설명하는 테스트 오라클의 종류를 쓰시오.

모든 입력 값들에 대하여 원한 결과를 생성하고 발생된 오류를 놓치지 않고 검출할 수 있는 오라클

15. 다음은 어떠한 프로토콜에 대한 설명이다.
보기를 읽고 답을 적으시오.

- 역순 주소 결정 프로토콜
- MAC을 이용해 IP를 검색
- 네트워크 계층에 속해서, 데이터 링크의 MAC을 이용해 네트워크 계층의 IP 검색

16. 다음이 설명하는 용어를 쓰시오.

네트워크 클래스는 IPv4 주소의 부족으로 인하여 각 사용 목적에 따라 주소 대역대를 할당하여 Network IP와 Host IP를 구분 짓는 방법이다. 이 중 (①)클래스는 국가 안 통신 네트워크에 적용하며, 첫 옥텟이 128로 시작하게 되는 통신 대역대이다.

17.

다음이 설명하는 용어를 쓰시오.

넷스케이프 커뮤니케이션즈 코퍼레이션에서 개발하였으며
HTTP의 보안이 강화되어 전자상거래 시 유용하게 사용되는
웹 통신 프로토콜

18.

다음 설명이 나타내는 용어를 쓰시오

시각, 활동, 지각, 청각 등에 장애를 가진 일부 사용자들이 다른 사람들과 동등하게 웹 탐색을 하고, 컴퓨터를 사용하기 위해 이용되는 특수한 장치 및 기술의 설정을 할 수 있는 기능의 이름. 이전 Windows의 버전에서는 '내게 필요한 옵션'이라는 이름이었으나 현재는 수정되었다.

19. 그래픽 유저 인터페이스(GUI)를 사용하는 Windows와 달리, 문자 기반의 사용자 인터페이스로 명령어를 사용하는 DOS의 인터페이스 방법은?

20. 윈도우 운영체제에서 시스템 매개변수 및 프로세스 종류 등을 기록한 저장소로 하드웨어, 운영체제 소프트웨어 등에 대한 정보와 설정이 들어있는 것을 뜻하는 단어는?

1.	2
2.	1 1 1 0
3.	15 17 f
4.	DROP COLUMN
5.	MSSQL
6.	ON
7.	독립성
8.	DBA
9.	DDL
10.	외래키

11.	WHERE
12.	블랙박스 테스트
13.	① : 하향식 테스트 ② : 상향식 테스트
14.	참 오라클
15.	RARP
16.	B
17.	HTTPS
18.	접근성
19.	CUI
20.	레지스트리(Registry)

PATH 01**1. 이분검색(Binary Search)**

- 찾고 싶은 데이터와 시작 위치에 대한 값, 종료 위치 값과 중간값을 이용해 배열 내에서 검색하는 방식
- 배열을 반으로 나누고, 나뉜 두 범위에서 찾고 싶은 데이터가 속한 범위를 찾은 후 일련의 행동을 반복한다.

답: 2**PATH 02****1. 진법 변환(10진법 → 2진법)**

$$- 1_{(10)} = 0000\ 0001_{(2)}, 2 = 0000\ 0010, 3 = \begin{array}{ccccccc} 128 & 64 & 32 & 16 & 8 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$

2. 코드 분석

```
while(i>0) {
    ar[j] = i%2; // 2로 나눈 나머지(2진수)를 배열에 입력
    i = i/2;      // 나누고 다음에 사용할 뜻을 i에 지정
    j++;
}

while(j>0) {
    j--;          // 배열에 입력된 2진수를 거꾸로 출력
    printf("%d ",ar[j]);
}
```

답: 1110

PATH

PATH 03

1. 진수 변환 (2진수, 8진수, 10진수, 16진수)

- 프로그래밍에서는 기본적으로는 10진 형태로 숫자를 사용합니다.
- 다른 진수로 표현하기 위해서는 0b, 0, 0x와 같은 문자를 접두사로 표현해야 합니다.

ex) 2진수 = 0b1001

8진수 = 067

↳ 파이썬 등의 언어에서는 0o 로 사용

16진수 = 0xA2

2. 진수 입,출력 (10진수, 8진수, 16진수)

- scanf 나 printf에서 사용하는 진수 사용법으로는

ex) 10진수 = %d

8진수 = %o

16진수 = %x

```
int num = 0b1111; // 2진수 1111 입력
```

답 : 15

```
printf("%d\n", num); // 10진수 출력 15
printf("%o\n", num); // 8진수 변환 출력 17
printf("%x\n", num); // 16진수 변환 출력 f
```

17
f

PATH 04

1. ALTER TABLE

- ADD : 컬럼 추가

ALTER TABLE 테이블명

ADD 컬럼명 데이터 유형;

- DROP COLUMN : 컬럼 삭제

ALTER TABLE 테이블명

DROP COLUMN 컬럼명;

- MODIFY COLUMN : 제약조건 변경

ALTER TABLE 테이블명

MODIFY 컬럼명 데이터 유형;

- RENAME COLUMN : 컬럼명 변경

ALTER TABLE 테이블명

RENAME COLUMN 컬럼명 TO 새로운컬럼명

답 : **DROP COLUMN**

PATH 05**1. MSSQL**

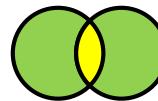
- 마이크로소프트(Microsoft)에서 1989년에 사이베이스(Sybase) 기반으로 개발된 RDBMS
- 오라클이나 MySQL과는 다르게 Window Server에서만 구동 가능

2. MySQL

- SQL을 사용하는 오픈 소스 관계형 데이터베이스 관리 시스템
- 오라클 사가 관리 및 지원
- 다중 사용자 및 다중 스레드 형식을 지원
- C++, java, PHP, Python 스크립트 등을 위한 응용 프로그램 인터페이스(API)를 제공
- 유닉스나 리눅스, 윈도우 운영 체제 등에서 사용 가능

답 : MSSQL**PATH 06****1. INNER JOIN(내부조인)**

- 두 테이블에서 조인된 필드가 일치하는 행만을 표시하는 조인



INNER JOIN

2. INNER JOIN문

```
SELECT *(or 컬럼명)  
FROM 테이블1  
INNER JOIN 테이블2  
ON 테이블1.컬럼명 = 테이블2.컬럼명;
```

답 : ON

PATH 07**1. 데이터베이스 트랜잭션(Transaction)**

- 하나의 작업을 수행하기 위해 필요한 연산들의 집합
- 데이터베이스에서 논리적인 작업의 단위
- **트랜잭션의 특성(ACID)**

원자성	일관성	독립성	지속성
Atomicity	Consistency	Isolation	Durability

답 : 독립성**PATH 08****1. DBA(DataBase Administrator)**

- 데이터베이스 관리자

2. DBMS(DataBase Management System)

- 관리자가 데이터베이스를 관리하는 시스템/프로그램

답 : DBA

PATH

PATH 09

데이터베이스 언어인 SQL(Structured Query Language)는
정의어(DDL), 조작어(DML), 제어어(DCL)로 구성되어 있습니다.

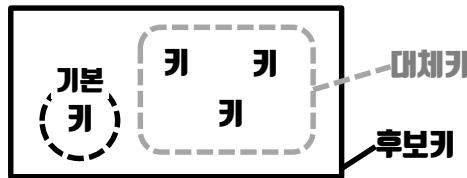
정의어 DDL(Database Definition Language)는
DBMS 내에서 데이터베이스의 전체 골격을 결정합니다.

대표적인 명령으로 CREATE(테이블 생성), ALTER(테이블 수정),
DROP(테이블 삭제), TRUNCATE(테이블 초기화) 등이 있습니다.

답 : DDL

PATH 10

키(Key)



1. **후보키(Key)** : 유일성 / 최소성 만족
2. **기본키** : 후보키 중 선정된 키로 중복 값 입력이 불가능하다.
Null 값을 가질 수 없다.
3. **슈퍼키** : 유일성 만족 / 최소성 불만족
4. **대체키** : 기본 키로 선택되지 못한 후보 키
5. **외래키** : 다른 테이블의 행을 식별하는 키

답 : 외래키

PATH 11**1. WHERE**

- 특정 레코드에 대한 조건을 설정할 때 사용되는 구문으로, 주로 DML의 명령어와 같이 사용된다.

2. DML

- 검색

```
SELECT column_name FROM table_name
```

- 삭제

```
DELETE column_name FROM table_name
```

- 갱신

```
UPDATE table_name
```

```
SET column1 = value1, column2 = value2...
```

- 추가

```
INSERT INTO table_name(col1,col2...)  
VALUE (value1,value2...)
```

답 : WHERE**PATH 12****1. 블랙박스 테스트**

- **사용자 관점**(내부 구조나 작동원리를 모르는 상태)으로 **명세**(요구사항과 결과물의 일치)기반의 테스트 방법입니다.

종류

- 균등분할(동치분해)
- 원인효과그래프테스트
- 한계값(경계값)테스트
- 비교테스트

2. 화이트박스 테스트

- **개발자 관점**으로 내부구조와 동작을 테스트합니다.

종류

- 기초경로테스트
- 루프테스트
- 제어흐름테스트
- 데이터흐름테스트
- 조건테스트
- 분기테스트

답 : 블랙박스 테스트

PATH 13

1. 통합 테스트

	빅뱅(Big Bang)	상향식(Bottom Up)	하향식(Top Down)
수행방법	모든 모듈을 동시 통합후 수행	최하위 모듈부터 점진적으로 상위 모듈과 함께 수행	최상위 모듈부터 하위 모듈들을 통합하며 수행
드라이버 /스텝	X	드라이버 필요	스텝 필요
장점	단시간 테스트 가능 작은 시스템에 유리	장애 위치 파악 쉬움 모듈개발 시간낭비가 없음	장애 위치 파악 쉬움 이른 프로토타입 가능 중요 모듈의 선 테스트 가능 결함 조기 발견 가능
단점	장애 위치 파악 어려 움 모든 모듈 개발	이른 프로토타입 어려움 중요 모듈이 마지막으로 테스트될 가능성 높음	많은 스텝이 필요 하위 모듈들의 불충분한 테스트 수행

답 : ① - 하향식 테스트
② - 상향식 테스트

PATH 14

1. 오라클

종류	정의
참 (True)	모든 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 발생된 모든 오류를 검출할 수 있음
샘플링 (Sampling)	특정한 몇몇 테스트 케이스의 입력 값들에 대해서만 기대하는 결과를 제공
추정 (Heuristic)	샘플링 오라클을 개선한 오라클로, 특정 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 나머지 입력 값들에 대해서는 추정으로 처리
일관성 검사 (consistent)	애플리케이션의 변경이 있을 때, 테스트 케이스의 수행 전과 후의 결과 값이 동일한지를 확인

답 : 참 오라클

PATH 15

1. ARP

- 주소 재해석 및 주소 결정 프로토콜이라 불립니다.
- 논리적인(IP) 주소를 이용해 물리적인(MAC) 주소를 검색합니다.
- 네트워크 계층에 속해있으며, 데이터 링크 계층의 MAC 주소와 연결해 줍니다.

2. RARP

- 역순 주소 결정 프로토콜이라 불립니다.
- 물리적인(MAC) 주소를 이용해 논리적인(IP) 주소를 검색합니다.
- 네트워크 계층에 속해있지만, ARP와는 다르게 데이터 링크 계층의 MAC 주소를 이용해 네트워크 계층의 IP를 검색해 줍니다.

답 : RARP

PATH 16

1. 네트워크 클래스(Classful network)

- 네트워크 단말의 증가로 가용 가능한 IPv4의 주소가 부족해졌고, 이에 사용 목적에 따라 IP의 대역대를 나누어 각 규모에 따라 관리하기 쉽게 표현한 것을 “클래스”라 합니다.

Class	IP주소의 첫째 옥텟	사용 목적	이론적 IP 주소 범위
A class	0xxx xxxx	대륙 안 통신	0.0.0~127.255.255.255
B class	10xx xxxx	국가 안 통신	128.0.0.0~191.255.255.255
C class	110x xxxx	기업 안 통신	192.0.0.0~223.255.255.255
D class	1110 xxxx	그룹 통신	224.0.0.0~239.255.255.255
E class	1111 xxxx	연구용 통신	240.0.0.0~255.255.255.255

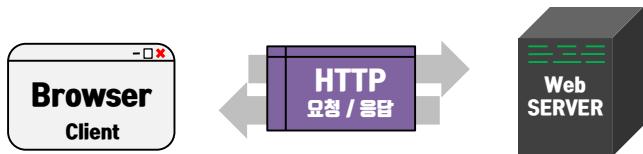
답 : B

PATH 17

1. HTTP(Hyper Text Transfer Protocol),

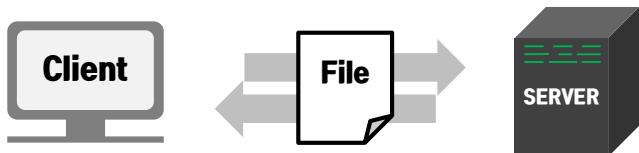
HTTPS(Hyper Text Transfer Protocol over Secure Socket Layer)

- WWW 상에서 클라이언트(Client)와 서버(Web Server) 간에 데이터 전송 시 사용



2. FTP(File Transfer Protocol)

- 클라이언트와 서버 간에 파일을 전송하기 위해 사용



답 : HTTPS

PATH 18

1. 접근성

- 돋보기, 고대비, 내레이터 등 어떠한 제한 때문에 사용하기 불편하거나 사용할 수 없을 때를 대비하기 위한 기능
- 컴퓨터 용어로써만 한정되는 것이 아닌, 산업 디자인과 건축 등의 분야에서도 많이 사용되는 용어로, 사용자들의 신체적 특성 혹은 나이나 지식 수준 등을 고려하여 가능한 많은 사용자가 이용할 수 있도록 하는 것.

답 : 접근성

PATH 19**1. GUI(Graphical User Interface)**

- 그래픽 기반의 사용자 인터페이스 / 아이콘 사용
- CUI 대비 직관적인 조작이 가능
ex> Windows

2. CUI (Character User Interface) / CLI

- **문자 기반의 사용자 인터페이스** / 명령어 사용
- GUI 대비 리소스의 소비가 적음
ex> DOS, Unix, Linux

답 : CUI**PATH 20****1. 레지스트리 (Registry)**

- 프로세스종류, 주기억장치의 용량, 주변장치의 정보, 시스템 매개변수, 응용소프트웨어에서 취급하는 파일 타입과 매개변수 등을 기록한 저장소이다.
- 모든 하드웨어, 운영 체제 소프트웨어, 대부분의 비운영 체제 소프트웨어, 사용자 PC 선호도 등에 대한 정보와 설정이 들어 있다

답 : 레지스트리(Registry)

PATH

**정보처리기능사 실기
예상문제 14회**

- 01.** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int a=45, b=3, c=3, d=2;
    a/= b-- + c * d;
    printf("%d",a);
}
```

- 02.** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int a, b = 10;
    for(a=0;a<5;a++){
        b -= a;
    }
    printf("%d, %d",a,b);
}
```

- 03.** 다음 C 언어로 구현된 프로그램을 분석하여
괄호 안에 알맞은 코드를 작성하시오.

```
#include <stdio.h>

int main(void) {
    int num;
    scanf("%d",&num);

    if(( ① ) == 0)
        printf("%d=짝수입니다.\n",num);
    else
        printf("%d=홀수입니다.\n",num);

}
```

- 04.** 다음 C 언어로 구현된 프로그램을 분석하여
그 실행 결과를 쓰시오.

```
#include <stdio.h>
main(void) {
    int ar[5]={44,66,88,22,11};
    int i,j,temp;
    for(i=0;i<5;i++){
        for(j=0;j<5;j++){
            if(ar[i]<ar[j]){
                temp = ar[i];
                ar[i]=ar[j];
                ar[j]=temp;    }
        }
    for(i=0;i<5;i++){
        printf("%d ",ar[i]);
    }
}
```

05.

다음의 빈칸에 들어갈 DML 명령어를 작성하시오.

- 검색 : (①) 컬럼명 FROM 테이블명
- 갱신 : UPDATE 테이블명 SET 컬럼명=수정값
- 삭제 : DELETE [FROM] 테이블명
- 삽입 : (②) INTO 테이블명 VALUES 입력값1, 입력값2…

06.

다음 <영어> 테이블에서 등급이 A부터 D 사이인 학생의 이름을 검색하되 중복된 값을 제거 하는 구문이다.
빈칸에 들어갈 단어를 작성하시오.

```
SELECT ( ① ) 이름  
FROM 영어  
WHERE 등급 ( ② ) ('A', 'B', 'C', 'D');
```

07. 아래의 <보기>의 빈칸에 들어갈 키(Key)의 종류를 채워 넣으시오.

- (①) : 유일성 / 최소성 만족
- (②) : 선정된 후보 키, 중복 값과 NULL 값 불가
- (③) : 유일성 만족 / 최소성 불만족
- (④) : 기본 키가 아닌 후보 키

08. 다음은 DB 트랜잭션의 특성인 ACID에 대한 설명이다.
빈칸에 들어갈 알맞은 단어를 작성하시오.

- 원자성(Atomicity) : 실행 중 중단되지 않게 하는 능력
- 일관성(Consistency) : 실행의 완료 후 데이터베이스 상태를 유지
- (①)(Isolation) : 한 트랜잭션의 실행 중 다른 트랜잭션이 끼어들지 못하게 보장
- 지속성(Durability) : 실행에 성공한 트랜잭션은 영원히 반영

09.

다음에서 설명하는 용어를 쓰시오 .

- 릴레이션을 처리하기 위한 연산의 집합 및 절차적 언어
- SELECT, PROJECT, JOIN, DIVISION 등의 순수 관계 연산이 포함되어 있다.
- 합집합, 교집합, 차집합 등 일반 집합 연산이 포함되어 있다.

10.

다음 설명이 나타내는 용어를 쓰시오.

- (①) : 사물 또는 사건으로 정의되며 개체라고도 부름
- (②) : 개체가 가지고 있는 요소 또는 성질을 부름
- (③) : 두 개체 간의 관계를 정의

11. 다음이 설명하는 용어를 쓰시오.

부적절한 자료가 입력되어 동일한 내용에 대하여 서로 다른 데이터가 저장되는 것을 허용하지 않는 성질의 종류 중 하나로, 기본키는 널(NULL)값이나 중복값을 가질 수 없다는 것을 의미

12. 다음의 지문이 설명하는 단어는 무엇인가?

(①)은 시스템의 구조나 행위, 더 많은 부들을 정의하는 개념적인 모형이다. 시스템의 컴포넌트가 무엇인지와 각 어떻게 상호작용되며 정보가 어떻게 교환되는지에 대해 정의한다.

13. 상위 모듈이 없이 하위 모듈만 있을 때, 상위 모듈로 테스트를 진행하기 위한 상향식의 테스트 시 필요한 것을 무엇이라고 하는가?

14. 다음 설명 하는 용어를 작성하시오.

- 소프트웨어를 검토할 때, 정해진 개발자나 전문가가 아닌 참석자 혹은 검사관들이 사용자, 테스터 등의 서로 다른 입장에서 코드에 대한 검토를 한다.

15. 다음에서 설명하는 단어는 무엇인가?

단말과 단말을 이어주는 분기의 접합점 혹은 단말 그 자체를 부르는 말로, 이것들에 의해 자율적으로 구성된 기반 구조 없는 네트워크를 애드혹 네트워크(Adhoc Network)라 한다.

16. 다음에서 설명하는 프로토콜은 무엇인가?

- 데이터 전송시 오류가 발생할 경우 재전송하지 않는 UDP와는 다르게 속도는 느리지만 데이터 흐름 제어를 통해 높은 신뢰성의 데이터를 제공한다.

17. IPv4로 작성된 IP 192.0.1.122에 사용된 비트(bit) 수를 구하시오.

18. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오.

- 쉘은 (①)과 프로그램 사이에서 상호작용을 가능하게 해주는 명령어 해석기를 의미하며, 사용자로부터 명령을 받아 (①)이 이해하도록 해석하고 그에 대한 처리 결과를 실행시키는 역할을 수행한다.

19. 다음은 운영체제의 어떠한 기능에 관한 지문이다.
보기를 읽고, 빈칸에 들어갈 알맞은 단어를 작성하시오.

하나의 PC에 여러 개의 운영체제를 설치하여 서로 다른 운영체제
에서 구동되는 프로그램을 실행시키도록 지원하는 것 중 하나로,
하드웨어가 아닌 네트워크 등을 결합하거나 나누는 기능.

20. Windows 10에서 Windows 설정을 열 수 있도록 해주는 단축키는 무엇인가?

1.	5
2.	5 0
3.	num%2
4.	11 22 44 66 88
5.	① SELECT ② INSERT
6.	① DISTINCT ② IN
7.	① 후보키 ② 기본키 ③ 슈퍼키 ④ 대체키
8.	독립성
9.	관계 대수
10.	① 엔티티(Entity) ② 속성(Attribute) ③ 관계(Relationship)

11.	개체 무결성
12.	시스템 아키텍처 (System Architecture)
13.	드라이버
14.	인스펙션(Inspection)
15.	노드(Node)
16.	TCP
17.	32bit
18.	커널(Kernel)
19.	리소스 가상화
20.	Win + i

PATH 01**1. 코드 분석**

```

int a=45, b=3, c=3, d=2;

a/= b-- + c * d; // a = a / b-- + c * d;
// a = 45 / (3-- + 3 * 2);
// a = 45 / (3 + (3 * 2));
// a = 45 / 9;
printf("%d",a); // a = 5

```

2. 산술 연산자 우선 순위

- 프로그래밍 언어에서도 일반적으로 우리가 알고 있는 수학과 동일하게 연산자에 따른 우선순위가 결정됩니다.

우선 순위	1	2	3
연산자	(괄호)	* , /, %	+,-

우선순위에 의해 위 문제에서는

산술 연산자 연산 후 복합대입연산자가 연산 됩니다.

답: 5

PATH 02**1. 코드 분석**

```

int a, b = 10;

for(a=0;a<5;a++){
    // a = 0    b = 10
    // a = 1    b =  9
    b -= a;
    // a = 2    b =  7
    // a = 3    b =  4
    // a = 4    b =  0
    // 반복문 종료시 a 증가
}

printf("%d, %d",a,b); // a = 5  b = 0

```

답: 5 0

PATH 03

1. 코드 분석

```
scanf("%d",&num); // num 변수에 입력

if( num%2 == 0) // 0과 같을때 짝수 값이 출력 되므로
    printf("%d=짝수입니다.\n",num);
else
    printf("%d=홀수입니다.\n",num);
```

답 : num%2

PATH 04

1. 코드 분석

```
int ar[5]={44,66,88,22,11};
int i,j,temp;
for(i=0;i<5;i++){
    for(j=0;j<5;j++){           // ar[i] 보다 ar[j]가 클 경우
        if(ar[i]<ar[j]){       // 버블 정렬
            temp = ar[i];
            ar[i]=ar[j];
            ar[j]=temp;
        }
    }
}
```

배열

```
// i=0,j=0 -> ar[0]<ar[0] 불만족 j증가 44 66 88 22 11
// i=0,j=1 -> ar[0]<ar[1] 만족 ar[0]=66 66 44 88 22 11
// i=0,j=2 -> ar[0]<ar[2] 만족 ar[0]=88 88 44 66 22 11
// i=0,j=3 -> ar[0]<ar[3] 불만족 j증가 88 44 66 22 11
// i=0,j=4 -> ar[0]<ar[4] 불만족 j증가 88 44 66 22 11
// i=1,j=0 -> ar[1]<ar[0] 만족 ar[1]=88 44 88 66 22 11
// i=1,j=1 -> ar[1]<ar[1] 불만족 j증가 44 88 66 22 11
// i=1,j=2 -> ar[1]<ar[2] 불만족 j증가 44 88 66 22 11
...
...
```

디버깅은 끝까지 차근차근 !!

답 : 11 22 44 66 88

PATH 05**1. DML(Data Manipulation Language)**

- 데이터를 **검색(SELECT)**, **삽입(INSERT)**, **갱신(UPDATE)** 하는 등
저장된 데이터를 실질적으로 처리하는 조작 언어
- 종류 : SELECT, DELETE, INSERT, UPDATE문

사용법	구분
SELECT 컬럼명 FROM 테이블명	검색
UPDATE 테이블명 SET 컬럼명=수정값	갱신
DELETE [FROM] 테이블명	삭제
INSERT INTO 테이블명 VALUES 입력값1, 입력값2...	삽입

답 : ① - SELECT**② - INSERT****PATH 06****1. IN (A , B)**

- SQL 구문의 조건 문에서 지정한 값 중 하나라도 일치하는 것을 반환합니다.
ex) 학년 IN(1,2); → 학년이 1학년 이거나 2학년

2. BETWEEN A AND B

- SQL 구문의 조건 문에서 지정한 두 수 사이의 숫자를 반환합니다.
ex) 나이 BETWEEN 10 AND 50 → 10세부터 50세까지 사이

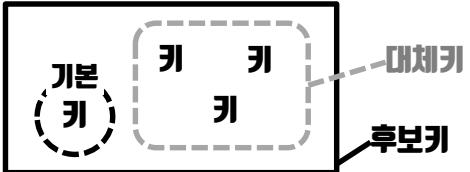
3. DISTINCT

- 데이터의 중복을 제거

답 : ① DISTINCT ② IN

PATH 07

키(Key)



1. 후보키(Key) : 유일성 / 최소성 만족
2. 기본키 : 후보키 중 선정된 키로 충복 값 입력이 불가능하다.
Null 값을 가질 수 없다.
3. 슈퍼키 : 유일성 만족 / 최소성 불만족
4. 대체키 : 기본 키로 선택되지 못한 후보 키
5. 외래키 : 다른 테이블을 연결해주는 다리와 같은 역할

답 : ① - 후보키 ② - 기본키
 ③ - 슈퍼키 ④ - 대체키

PATH 08

1. 데이터베이스 트랜잭션(Transaction)

- 하나의 작업을 수행하기 위해 필요한 연산들의 집합
- 데이터베이스에서 논리적인 작업의 단위
- 트랜잭션의 특성(ACID)

원자성	일관성	독립성	지속성
Atomicity	Consistency	Isolation	Durability

답 : 독립성

PATH 09**1. 관계 대수**

- 원하는 정보를 검색하기 위해서 정보를 어떻게 유도하는가를 연산자를 이용하여 정보를 구하는 **절차적 언어**.
- 종류 : SELECT, JOIN 등등

2. 관계 해석

- 관계 데이터의 연산을 표현하는 방법 중의 하나로, 원하는 정보가 무엇인지만을 명시하는 **비절차적 언어**.
- 종류 : 튜플 관계 해석, 도메인 관계 해석 등등

답 : 관계 대수**PATH 10****1. 엔티티(Entity)**

- 사물 또는 사건으로 정의되며 개체라고도 한다. ERD에서 엔티티는 사격형으로 나타내고 사격형 안에는 엔티티의 이름을 넣는다.

2. 속성(Attribute)

- 엔티티가 가지고 있는 요소 또는 성질을 속성이라 부른다. 속성은 선으로 연결된 동그라미로 표기하거나 표 형식으로 표기하기도 한다.

3. 관계(Relationship)

- 두 엔티티 간의 관계를 정의한다.

**답 : ① – 엔티티(Entity)
② – 속성(Attribute)
③ – 관계(Relationship)**

PATH 11**1. 무결성 종류**

- 개체 무결성
- 릴레이션에서 기본키를 구성하는 속성을 널(NULL)값이나 중복값을 가질 수 없다.
- 참조 무결성
- 외래키 값은 NULL이거나 참조 릴레이션의 기본키 값과 동일해야 한다.
즉 릴레이션은 참조할 수 없는 외래키 값을 가질 수 없다.
- 외래키와 참조하려는 테이블 기본키는 도메인과 속성 개수가 같아야 한다.
- 도메인 무결성
- 특정 속성의 값이, 그 속성이 정의된 도메인에 속한 값이어야 한다는 규정.

답 : 개체 무결성

PATH 12**1. 아키텍처(Architecture)**

- 시스템 아키텍처와 소프트웨어 아키텍처로 구분

2. 소프트웨어 아키텍처(Software Architecture)

- 소프트웨어의 골격이 되는 기본 구조
- 구성요소(Component) 간의 관계를 표현하는 시스템 구조/구조체
- 설계 기본원리 : 모듈화 / 추상화 / 단계적 분해 / 정보은닉

3. 시스템 아키텍처(System Architecture)

- 하드웨어와 소프트웨어 기반으로 시스템에서 서비스를 제공하기 위한 아키텍처
- **최적화를 목표로 시스템의 구성과 원리, 환경 등을 설명 및 설계**

**답 : 시스템 아키텍처
(System Architecture)**

PATH 13

구분	스텝	드라이버
시기	상위 모듈은 있지만 하위 모듈이 없을 때, 하위 모듈 로의 테스트	상위 모듈 없이 하위 모듈만 있을 때, 상위 모듈 로의 테스트
방식	하향식 테스트	상향식 테스트
특징	작성이 쉬운 시험용 모듈	존재하지 않는 상위 모듈간의 인터페이스 역할

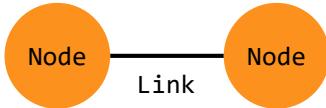
답 : 드라이버

PATH 14

1. 워크스루 와 인스펙션 비교

구분	워크스루	인스펙션
목적	산출물 평가 및 개선	결합 파악 및 제거
수행 조건	팀이나 관리자의 필요 시	완성도 기준 이상 시
결합 수정 여부	저자 결정	모든 결합 제거
검토자 인원	2 - 7 명	3 - 6 명
참여자	기술 전문가 및 동료	동료
검토 인도자	저자	진행자
검토 준비 여부	준비하지 않음	체크리스트 활용
검토 속도	빠름	느림
발표자	저자	리더
지표 수집 여부	수집하지 않음	수집함
보고서	워크스루 보고서	결합 리스트
데이터 측정 여부	권장	필수
체크리스트 사용 여부	사용하지 않음	사용함

답 : 인스펙션(Inspection)

PATH 15**1. 노드(Node)**

- 단말과 단말을 이어주는 분기와의 접합점 혹은 단말 그 자체

※ 개정 전 정보처리기능사/산업기사/기사 출제문제 경우

노드란 두 지점을 이어주는 분기 자체를 뜻하지만, 본 해설에서는 사전적 의미를 참고하여 개시하였습니다.

2. 애드혹 네트워크(Adhoc Network)

- 유선망이 고정적이지 않으면 이동 호스트로 이루어져,
유선망의 구성이 어렵거나 단기간 사용에 적합합니다.

답 : 노드(Node)

PATH 16**1. TCP(Transmission Control Protocol)**

- 데이터 흐름 제어를 통한 높은 신뢰성 제공
- 연결형 서비스의 가상 회선 방식
- 전이중(Full-Duplex), 점대점(Point to Point)

2. UDP (User Datagram Protocol)

- 흐름제어를 하지 않기 때문에 TCP보다 데이터 전송속도가 빠르다.
- TCP의 경우 한번에 모든 데이터를 읽을 수 있지만,
UDP는 여러 번의 데이터를 전송했다면, 여러 번을 읽어야 합니다.
- (데이터 경계가 존재)
- 송수신 중 데이터 누락/오류 발생 시, 재전송하지 않으며,
데이터의 순서를 유지하지 않는다.(비신뢰성 프로토콜)
- 연결 설정이 없는 비연결성 프로토콜이다.(1vs1, 1vs다수 등)

답 : TCP

PATH 17

구분	IPv4	IPv6
주소 길이	32bit	128bit
표시 방법	8비트씩 4부분 (10진수)	16비트씩 8부분 (16진수)
주소 개수	약 43억 개	무한대에 가깝게 사용 가능
주소 할당	A,B,C 등 클래스 단위의 비순차적 할당	네트워크 규모 및 단말기 수에 따른 순차적 할당
품질 제어	지원 수단 없음	등급별, 서비스별로 패킷 구분
보안 기능	Ipsec 프로토콜 별도 설치	확장 기능에서 기본으로 제공
PnP	지원 수단 없음	지원 수단 있음

답 : 32bit 또는 32

PATH 18



1. 쉘(Shell)

- 커널과 사용자 간의 대화를 가능하게 하는 명령어 해석기

2. 커널(Kernel)

- 운영체제의 핵심(중심)
- 소프트웨어와 하드웨어간의 상호 작용을 관리하는 프로그램
ex) 게임 실행 시, 게임(소프트웨어)에서 하드웨어의 성능을 조절

답 : 커널(Kernel)

PATH 19

1. 리소스 가상화

- 메모리, 저장장치, 네트워크 등을 결합하거나 나누는 것.
- 사용자는 가상화된 물리적 장치의 위치를 모르는 경우가 대부분

2. 플랫폼 가상화

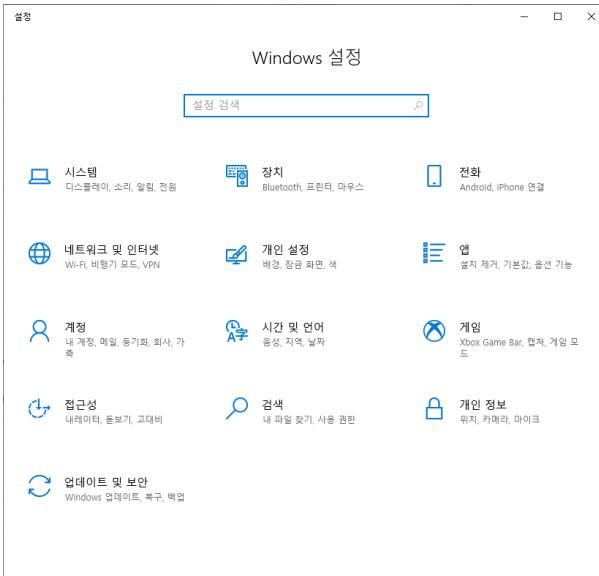
- 하드웨어 플랫폼 위에서 실행되는 호스트 프로그램이 게스트 프로그램을 만들어서 마치 독립된 환경인 것처럼 보여주는 것.

답 : 리소스 가상화

PATH 20

1. Windows 설정

- 시스템, 장치, 계정, 접근성 등의 Windows 10 혹은 컴퓨터 설정과 관련된 기능을 수정할 수 있는 곳



답 : Win + i

PATH

**정보처리기능사 실기
예상문제 15회**

01. 하나의 데이터 패킷을 전송하면 동일 네트워크 내에 있는 모든 장비에게 전송할 수 있는 네트워크는 무엇인가?

02. 다음 <보기>를 참조하여 빈칸에 알맞은 내용을 작성하시오.

토플로지란 네트워크 상의 장비 등을 연결하여 이루어진 네트워크 망의 형태를 의미한다.

토플로지의 형태 중 (①) 란 (②)의 변형된 형태로서, 하나의 1차 허브를 통해 더 많은 노드들을 연결하는 형태를 이루어 부모와 자식과 같은 관계를 갖고 있으며, 데이터의 양방향 통신을 통해 모든 노드에 전송이 가능하다.

(①)는 중앙집중형 구조를 가지고 있으며, 이로 인한 노드의 추가 등에 용이하나 중앙 노드에 장애가 발생하게 되면 전체 네트워크의 사용이 불가능해진다.

03.

다음 C 언어로 구현된 프로그램을 분석하여,
그 실행 결과를 쓰시오.(5점)

04.

정적 테스트의 일종으로 전문가들을 통해 소프트웨어의
작업 내역을 사전에 검토 후 회의를 진행하는 형태를 통해
오류를 검출하는 기법은 무엇인가?

```
#include <stdio.h>
#include <math.h>
int main(void) {
    char num1 = 24;
    char num2 = -25;
    char result = fabs(num1--) - fabs(++num2);

    printf("%.0f", result);
    return 0;
}
```

- 05.** 다음 <표>를 참조하여 테이블의 크기 변경되지 않은 상태로 점수가 85점 이상인 레코드를 삭제하는 명령어를 작성하시오.

< 학생 테이블 >

순번	성명	학과	점수	총점
1	김재연	컴퓨터공학	92	A
2	강민준	시각디자인	80	D
3	나시찬	정보통신	84	C
4	장지웅	경영학과	87	B

(①) FROM 학생 (②) 점수 >= 85

- 06.** 네트워크 상에서 인터넷과는 다르게 조직이나 단체 내부에서만 통신이 가능하도록 설계되어진 네트워크망을 의미하며 해당 소속이 아닌 사용자의 접속을 차단하는 네트워크 형태는?

- 07.** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>

int main(void) {
    int num1 = 20;
    int num2 = 3;
    while(num1!=0){
        num1 /= num2;
        printf("%d ", num1);
    }
    return 0;
}
```

- 08.** 데이터베이스 내의 릴레이션을 수정하기 위한 명령어는 무엇인가?

09.

다음의 <표>를 참고하여 아래의 명령어를 실행하였을 때 뷰에서 "이름" 필드에서 출력되는 결과값을 입력하시오.

<성적 테이블>

학년	이름	국어	영어	수학
1	이진아	90	85	76
2	김진수	85	73	74
3	박수한	78	85	88
2	김하나	80	85	62
3	김지은	55	87	92

```
CREATE VIEW 출석부
```

```
AS SELECT 이름
```

```
FROM 성적
```

```
WHERE 이름 LIKE '김%' AND 국어 > 80
```

10.

다음에서 설명하는 용어를 쓰시오.(5점)

데이터베이스 내에서 테이블이 서로 연관되어 있어, 기존보다 효율적으로 데이터를 관리를 할 수 있으며, 또한 정규화 수행이 가능하여, 데이터의 중복성을 최소화할 수 있는 관리 시스템이다.

11.

다음에서 설명하는 용어를 쓰시오.(5점)

()란 소프트웨어 내부 구조를 참조, 논리적 모든 경로를 테스트하여 예상값과 같이 실행되는가 확인을 하는 기법을 의미하며, 모듈 안의 작동을 직접 관찰하는 등의 작업을 수행한다. 대표적인 테스팅 기법으로는 데이터 흐름 테스트, 반복 테스트, 조건 테스트, 경로 테스트 등이 존재한다.

12.

다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.(5점)

```
import java.lang.Math;
class PATH
{
    public static void main(String[] args)
    {
        double num1 = 2.5678;
        double num2 = 3;
        System.out.println(Math.rint(num1) +
                           Math.rint(num2));
    }
}
```

- 13.** 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
import java.lang.Math;  
  
class PATH  
{  
  
    public static void main(String[] args)  
    {  
  
        int num1 = 25; int num2 = 13;  
  
        System.out.println(Math.max(num1, num2));  
  
        System.out.println(Math.min(num2, num1));  
    }  
}
```

- 14.** Windows의 ipconfig와 같이 IP, MAC 주소, 게이트웨이 등의 네트워크 환경을 확인할 수 있도록 도와주는 UNIX / LINUX 명령어를 작성하시오.

- 15.** 다음은 C언어로 구현된 프로그램으로
빈칸에 알맞은 결과 값을 입력하시오.(5점)

```
#include<stdio.h>

int main(void)
{
    int num1 = 128;
    int num2 = 0;

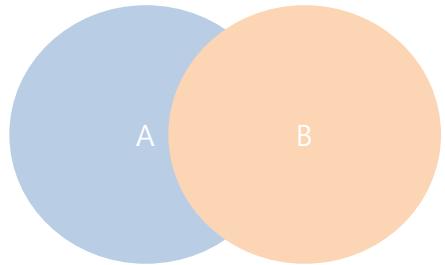
    printf("%d", num1^num2);

    return 0;
}
```

- 16.** 다음의 <보기>에서 설명하는 Windows 기능에 대해
작성하시오.

Windows의 제어판의 기능 중에 하나로서,
디스플레이가 없는 컴퓨터에서 사용할 경우 내레이터 기능을
사용하여 텍스트를 읽어주거나, 특정 입력장치가 없는 경우
다른 장치 등을 사용하여 이를 보완해주는 기능 등등
이외에도 돋보기, 고대비 설정 등을 지원해주는 기능이다.

17. 다음 <표>를 참조하여 다음에서 설명하는 용어를 작성하시오.



A의 데이터를 기준으로 B의 데이터를 제외한 나머지를 검색하기 위한 데이터베이스 집합 연산자이다.

18. 데이터베이스 제어어(DCL)에 속하며, 사용자에게 데이터에 대한 권한을 부여하는 명령어는 무엇인가?

19.

다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오.

```
#include <stdio.h>
main(void) {
    int ar[3][2]={{10,20},{30,10},{20,20}};
    int i,j,sum=0;
    for(i=0;i<3;i++)
    {
        for(j=0;j<2;j++)
        {
            sum+=ar[i][j];
        }
    }
    printf("%d\n",sum/(i+j));
}
```

20.

UNIX / LINUX 상에서 파티션 테이블을 관리할 수 있는 명령어로서, 하나의 볼륨에 파티션의 생성, 수정, 삭제 등의 구성을 할 수 있는 것은?

1.	BMA (Broadcast Multi-Access)
2.	① 계층형 or 트리형 ② 성형 or 스타형
3.	0
4.	워크스루
5.	① DELETE ② WHERE
6.	인트라넷(IntraNet)
7.	6 2 0
8.	ALTER
9.	김진수
10.	RDBMS

11.	화이트박스 테스트 (White-Box Testing)
12.	6.0
13.	25 13
14.	ifconfig
15.	128
16.	접근성 센터
17.	MINUS
18.	GRANT
19.	38
20.	fdisk

PATH 01**1. 멀티 액세스 네트워크 (Multi-Access)**

- 하나의 인터페이스를 통해 다수의 장비와 연결되는 네트워크

2. BMA(Broadcast Multi-Access)

- 하나의 브로드캐스트 패킷을 전송하면 동일 네트워크 내에 있는 모든 장비에게 도달할 수 있는 네트워크

3. NBMA (Non-Broadcast Multi-Access)

- 브로드캐스트가 없는 다중 접속 환경으로서 가상회선을 사용하는 네트워크

답 : BMA (Broadcast Multi-Access)

PATH 02**네트워크 토폴로지(topology)**

: 네트워크 상의 링크 및 노드들로 이루어진 망 구성 형태

1. 성형(or Star형)

- 중앙에 위치한 메인 노드를 통하여 다른 노드들과 통신하는 형태 (중앙 집중형)

**2. 계층형(or Tree형)**

- 1차 노드를 통해 더 많은 노드들을 연결하는 형태

**3. 버스형(Bus)**

- 하나의 파이프라인에서 각 노드들이 연결된 형태

**4. 링형(Ring)**

- 각 노드들이 인접한 노드들과 좌우로 연결된 형태

**5. 망형(Mesh)**

- 모든 노드들이 그물과 같은 형태로 얹혀 있는 망 형태

답 : ① 계층형 or 트리형 ② 성형 or 스타형

PATH 03

절대값 함수

함수 : double fabs(값)

헤더 파일 : math.h

ex) printf("%f", fabs(-15)) → 15.000000

함수 : int abs(값)

헤더파일 : stdlib.h

ex) printf("%d", abs(-15)) → 15

코드 분석

```
char num1 = 24; char num2 = -25;  
  
// num1 : 24 , num2는 -25 대입  
  
char result = fabs(num1--) - fabs(++num2);  
  
// fabs(24--) - fabs(++ -25)  
→ fabs(24) - fabs(-24)  
→ 24 - 24 = 0 출력
```

답 : 0

PATH 04

정적 테스트

: 프로그램을 직접 실행하지 않고, 명세나 소스 코드를 이용하여 분석하는 테스트 기법

1. 워크스루

- 전문가들을 통해 소프트웨어의 작업 내역을 검토하는 작업
(오류의 조기 검출)

2. 인스펙션

- 개발 단계에서의 내용물의 품질을 평가 및 개선하는 작업

답 : 워크스루

PATH 05**1. DELETE**

- 테이블을 삭제하는 것이 아닌 원하는 데이터(레코드)를 삭제할 수 있는 명령어
- 테이블의 크기 등은 변경되지 않는다.

2. DROP

- 테이블 전체를 삭제하는데 사용하는 명령어

3. TRUNCATE

- DELETE와 비슷하게 테이블을 삭제하지 않고 데이터를 지우지만 기존의 데이터 공간은 삭제되게 된다.

순번	성명

DELETE

순번	성명

TRUNCATE



DROP

답 : ① DELETE ② WHERE**PATH 06****인트라넷(IntraNet)**

- 기업이나 기관 내에서 내부 사용자들만이 사용할 수 있도록 만든 인터넷

엑스트라넷(ExtraNet)

- 인트라넷(IntraNet)의 확장된 개념
- 기업과 기업 간 혹은 기업과 협력사들만 통신하도록 설계되어진 내부 인터넷

답 : 인트라넷(IntraNet)

PATH 07**복합 대입 연산자**

- 대입 연산자와 다른 연산자를 줄여서 같이 쓰기 위한 연산자

< num은 5라고 가정 >

문법	예시	결과
<code>+=</code>	<code>num+=2</code> $\rightarrow \text{num} = \text{num} + 2$	7
<code>-=</code>	<code>num-=2</code>	3
<code>*=</code>	<code>num*=2</code>	10
<code>/=</code>	<code>num/=2</code>	2.5
<code>%=</code>	<code>num%=2</code>	1
<code>&=</code>	<code>num&=2</code>	0
<code> =</code>	<code>num =2</code>	7
<code>^=</code>	<code>num^=2</code>	7

답 : 620

PATH 08**DDL(Database Definition Language)**

: 데이터베이스 정의 언어, 테이블 등의 데이터 구조를 생성, 수정, 삭제 등을 하는 역할을 수행

종류	기능
CREATE	데이터베이스 내의 릴레이션(테이블)을 생성
DROP	데이터베이스 내의 릴레이션(테이블)을 삭제
ALTER	데이터베이스 내의 릴레이션(테이블)을 수정
TRUNCATE	릴레이션을 초기화(테이블의 크기 변경)

답 : ALTER

PATH 09**1. 뷰(View)**

- 하나 이상의 기본 테이블로부터 유도된 가상의 테이블을입니다.
- 구조와 조작도 기본 테이블과 매우 유사합니다.

2. 뷰 생성

- 대표적인 구문의 형식은 아래와 같습니다.

```
CREATE VIEW 뷰이름 AS
SELECT 컬럼1, 컬럼2, ...
FROM 테이블
WHERE 조건문;
```

TIP) 다중 조건 사용

AND 조건 : 모든 조건이 TRUE 일 때, 값을 출력

OR 조건 : 조건 중 하나라도 TRUE 일 때, 값을 출력

답 : 김진수**PATH 10****1. DBMS(DataBase Management System)**

- 사용자 및 관리자와 시스템 사이에서 데이터를 생성 및 관리할 수 있도록 도와주는 소프트웨어
- 정규화를 수행할 수 없어 데이터의 중복성을 피할 수 없음
- 데이터베이스 보안성이 낮음

2. RDBMS(Relational DataBase Management System)

- 관계형 데이터베이스 관리 시스템
- 테이블 간에 서로 연관되어 기존보다 효율적으로 데이터를 관리할 수 있다.
- 정규화를 실행하여 데이터에 대한 중복을 최소화할 수 있음

3. DBA(DataBase Administrator)

- 데이터베이스 관리자

답 : RDBMS

PATH 11**화이트 박스 검사 (White-Box Testing)**

- 응용 프로그램의 내부 구조와 동작을 검사하는 테스트
- 개발자가 코드까지도 확인하여 테스트 및 동작의 작동 유무를 직접 관찰.

종류

1. 데이터 흐름 테스트 (Data Flow Tesing)
2. 반복 테스트(Loop Testing)
3. 조건 테스트 (Condition Testing)
4. 경로 테스트 (Basic Path Tesing)

**답 : 화이트박스 테스트
(White-Box Testing)**

PATH 12**1. 코드 분석**

```
import java.lang.Math; // 수학 관련 클래스

double num1 = 2.5678;
double num2 = 3;
System.out.println(Math.rint(num1) +
Math.rint(num2));

// Math.rint는 값을 가장 가까운 인수로 반환하는 함수
// ex ) Math.rint(2.3) → 2.0 Math.rint(2.6) → 3.0

// Math.rint(2.5678) → 3.0
Math.rint(3) → 3.0
3.0 + 3.0 → 6.0 출력
```

답 : 6.0

PATH 13

1. 코드 분석

```

import java.lang.Math; // 수학 관련 클래스

int num1 = 25; int num2 = 13;
System.out.println(Math.max(num1, num2));
// Math.max는 최대값을 구하는 함수
→ 25 출력

System.out.println(Math.min(num2, num1));
// Math.min는 최소값을 구하는 함수
→ 13 출력

```

답 : 25**13****PATH 14**

1. UNIX / LINUX

- CLI (Command Line Interface) / CUI (Character User Interface)
기반의 텍스트를 통한 운영체제의 기능을 사용하는데 특화된 운영체제

2. MS-DOS / WINDOWS의 CMD(명령 프롬프트)

- DOS의 경우 CLI, WINDOWS는 GUI (Graphical)를 지원.
- WINDOWS의 경우 CMD를 이용하여 CLI 기능을 사용할 수 있다.

3. 명령어 비교

	UNIX	MS-DOS
네트워크 환경 확인	ifconfig	ipconfig
파일 및 디렉터리 복사	cp	copy
화면 지우기	clear	cls
현재 작업중인 위치 확인	pwd	dir

TIP) Windows에서 copy는 해당 파일을 복사하나
xcopy의 경우 하위 디렉토리 및 파일도 복사할 수 있다.

답 : ifconfig

PATH 15**1. 비트연산자**

- 10진수의 정수 값을 2진수로 변환하여 논리 연산을 하는 기능

문법	기능	예시
& (and)	이진수로 변환 값이 서로 같을 경우 1을 반환	num = 4 & 2
(or)	이진수로 변환 값이 둘 중에 하나라도 1이면 1을 반환	num = 4 2
^ (xor)	이진수로 변환 값이 서로 다를 경우 1을 반환	num = 4 ^ 2

2. 코드분석

```

num1 = 128;           // 128 : 1000 0000
num2 = 0;             // 0 : 0000 0000
printf("%d", num1 ^ num2);
^(xor) : 서로 다른 경우 1을 반환
// 1000 0000 ► 128
    
```

답 : 128**PATH 16****접근성 센터**

- 고대비 설정, 화상 키보드, 돋보기, 내레이터 기능 등을 지원하며, 특정 상호작용 기기를 대신하여 다른 장치를 이용하여 이를 보완할 수 있도록 도와주는 기능

답 : 접근성 센터

PATH 17

집합연산자

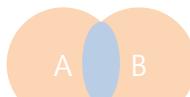
- UNION(합집합: 중복제거) : 중복을 제거한 결과의 합을 검색



- UNION ALL(합집합: 중복포함)



- INTERSECT(교집합) : A와 B가 겹치는 데이터의 행을 검색



- MINUS(차집합) : A의 데이터를 기준으로 B의 데이터를 제외한 나머지

지를 검색



답 : MINUS

PATH 18

SQL(Structured Query Language)의 구성

1. 데이터 정의어(DDL)
2. 데이터 조작어(DML)
3. **데이터 제어어(DCL)**

DCL(Database Control Language)

: DBMS 내에서 데이터의 보안, 무결성 등을 제어합니다.

1. COMMIT(작업완료)
2. ROLLBACK(작업취소)
3. GRANT(권한부여)
4. REVOKE(권한해제)

답 : GRANT

PATH 19

1. 코드 분석

```
int ar[3][2]={{10,20},{30,10},{20,20}};
int i,j,sum=0;
for(i=0;i<3;i++)
{
    for(j=0;j<2;j++)
    {
        sum+=ar[i][j]; //2차원배열 모든 합 저장 110
    }
}                                // 반복문 종료시 i=3, j=2
printf("%d\n",sum/(i+j)); // 산술연산자 우선순위에 의해
                           // 110/3 연산후 2 더함
```

답 : 38**PATH 20**

파티션이란?

- 하나의 물리적인 디스크에 여러 개의 논리적으로 나눈 구역을 의미

물리적인 디스크

DISK : 1TB

단일 파티션 : 하나의 디스크를 하나의 공간으로 사용

1TB

다중 파티션 : 하나의 디스크를 두 개 이상의 공간으로 사용

450 GB

550 GB

Tip) UNIX / LINUX에선 fdisk 를 통해 파티션 구성을 하며

Windows 상에서의 파티션 구성 방법은 diskpart 를 사용합니다.

답 : fdisk