

2018년 서울시 9급 컴퓨터일반 풀이

by 호이호이꿀떡

정답 체크

01	02	03	04	05	06	07	08	09	10
④	④	①	②	①	③	①	②	②	③
11	12	13	14	15	16	17	18	19	20
④	②	①	④	②	②	③	③	③	①

1. 중위 표기법으로 표현된 <보기>의 수식을 후위 표기법으로 옳게 표현한 것은?

< 보기 >

$$a + (b * c - d) * (e - f * g) - h$$

- ① $a b * c d + e f g * - * - h -$
- ② $a b c * d + e f * g - * - h -$
- ③ $a b c d * - e f g * + * - h -$
- ④ $a b c * d - e f g * - * + h -$

답 ④

$$a + (b * c - d) * (e - f * g) - h$$

$$\rightarrow a + (bc * - d) * (e - fg *) - h$$

$$a + (bc * - d) * (e - fg *) - h$$

$$\rightarrow a + (bc * d -) * (efg * -) - h$$

$$a + (bc * d -) * (efg * -) - h$$

$$\rightarrow a + (bc * d - efg * -) - h$$

$$a + (bc * d - efg * -) - h$$

$$\rightarrow (abc * d - efg * - * +) - h$$

$$\rightarrow abc * d - efg * - * + h -$$

2. 소프트웨어 개발 프로세스 모델에 대한 설명으로 가장 옳지 않은 것은?

- ① 폭포수(Waterfall) 모델은 단계별 정형화된 접근 방법 및 체계적인 문서화가 용이하다.
- ② RAD(Rapid Application Development) 모델은 CASE(Computer Aided Software Engineering) 도구를 활용하여 빠른 개발을 지향한다.
- ③ 나선형(Spiral) 모델은 폭포수(Waterfall) 모델과 원형(Prototype) 모델의 장점을 결합한 모델이다.
- ④ 원형(Prototype) 모델은 고객의 요구를 완전히 이해하여 개발을 진행하는 것으로 시스템 이해도가 높은 관리자가 있는 경우 유용하다.

답 ④

- ④ 프로토타이핑(prototyping) 모델(= 원형 모델)
요구 분석이 정확하지 않을 때, 개발의 일부분만을 원형(프로토타입, prototype)으로 개발하여 사용자에게 제공하여 시험 사용하게 한 뒤, 이를 통해 요구를 분석하고 점검평가·개선 작업을 통해 개발하는 모델이다.
요구를 완전히 이해한 상태에서 개발을 진행하는 것은 폭포수 모델 등 고전적인 모델이다.

- <오답 체크> ① 폭포수 모델(waterfall model)
단계별 순차적인 과정을 통한 개발론으로, [타당성 검토 - 계획 - 요구 분석 - 설계 - 구현 - 시험 - 유지보수]의 단계를 가진다.
- ② RAD(Rapid Application Development) 모델
짧은 개발주기(60일~90일) 동안 소프트웨어를 개발하기 위한 순차적 프로세스 모델로서, 빠른 개발을 위해 CASE 도구 사용
 - ③ 나선형 모델(Spiral Model, 점진적 모형)
폭포수 모형과 프로토타입 모형의 장점에 위험 분석 기능을 추가한 모델
각 단계별 나선 모양을 그려며 반복 수행한다.
 1. 요구 사항 및 위험 분석
 2. 계획 및 건축가 반복
 3. 구현
 4. 테스트 및 확인

3. 서로 다른 시스템 간의 통신을 위한 표준을 제공함으로써 통신에 방해가 되는 기술적인 문제점을 제거하고 상호 인터페이스를 정의한 OSI 참조 모델의 계층에 대한 설명으로 가장 옳지 않은 것은?

- ① 네트워크 계층은 물리 계층에서 전달받은 데이터에 대한 동기를 확인하는 기능, 데이터의 원활한 전송을 제어하는 흐름제어(Flow Control) 기능, 안전한 데이터 전송을 위한 에러 제어(Error Control) 기능을 수행한다.
- ② 물리 계층은 상위 계층으로부터 전달받은 데이터의 물리적인 링크를 설정하고 유지, 해제하는 기능을 담당한다.
- ③ 전송 계층은 통신하고 있는 두 사용자 사이에서 데이터 전송의 종단 간(end-to-end) 서비스 질을 높이고 신뢰성을 제어하는 기능을 담당한다.
- ④ 응용 계층은 사용자가 직접 접하는 부분이며 전자 메일 서비스, 파일 전송 서비스, 네트워크 관리 등이 있다.

답 ①

- ① 물리 계층에서 전달받은 데이터를 확인하고 오류를 감지하는 기능은 OSI 2계층 데이터 링크 계층에서 수행하다.
네트워크 계층은 OSI 3계층으로 경로 설정, 주소 변환, 논리 주소 지정, 흐름제어, 오류제어 등의 기능을 수행한다.

4. <보기> C 프로그램의 실행 결과는?

```

< 보 기 >
#include <stdio.h>

int main()
{
    int a=0, b=1;
    switch(a)
    {
        case 0 : printf("%d\n", b++); ; break;
        case 1 : printf("%d\n", ++b); ; break;
        default : printf("0\n", b); ; break;
    }
    return 0;
}

```

- ① 0
- ② 1
- ③ 2
- ④ 3

답 ②

switch문의 조건으로 a를 사용하기 때문에, case 0에 해당하는 첫 번째 문장을 실행한다. printf("%d\n", b++)에서 b++이므로, 먼저 b값 1을 출력을 한 다음에 b값을 증가시킨다.

5. 정책 수립에 있어 중요성이 커지고 있는 빅데이터에 대한 설명으로 가장 옳지 않은 것은?

- ① 디지털 환경에서 생성되는 데이터로 규모가 방대하고, 생성 주기가 길며, 형태가 다양하다.
- ② 하둡(Hadoop)과 같은 오픈 소스 소프트웨어 시스템을 빅데이터 처리에 이용하는 것이 가능하다.
- ③ 보건, 금융과 같은 분야의 빅데이터는 사회적으로 유용한 정보이나 데이터 활용 측면에서 프라이버시 침해에 대한 대비가 필요하다.
- ④ 구글 및 페이스북, 아마존의 경우 이용자의 성향과 검색 패턴, 구매패턴을 분석해 맞춤형 광고를 제공하는 등 빅데이터의 활용을 증대시키고 있다.

답 ①

① 빅데이터 환경에서 데이터 생성 주기는 점점 빨라지고 있다.
 <오답 체크> ② 하둡(Hadoop, High-Availability Distributed Object-Oriented Platform)
 아파치 하둡(Apache Hadoop). 분산 컴퓨터 환경에서 대량의 자료를 저장하고 처리할 수 있는 자바 기반의 오픈 소스 프레임워크이다.

6. <보기>는 TCP/IP 프로토콜에 대한 설명이다. ㉠~㉣에 들어갈 내용으로 가장 옳은 것은?

〈 보 기 〉

- (㉠)는 사용자가 입력한 IP 주소를 이용해 물리적 네트워크 주소(MAC Address)를 제공한다.
- (㉣)는 데이터 전송 과정에서 오류가 발생하면 오류 메시지를 전송한다

- | | | | |
|--------|------|--------|------|
| ㉠ | ㉡ | ㉢ | ㉣ |
| ① ICMP | RARP | ② RARP | ICMP |
| ③ ARP | ICMP | ④ ICMP | ARP |

답 ③

㉠ **ARP**(Address Resolution Protocol, 주소 결정 프로토콜)
 IP 주소를 대응하는 MAC 주소값으로 변환해주는 프로토콜

㉡ **ICMP**(Internet Control Message Protocol, 인터넷 제어 메시지 프로토콜)
 호스트와 게스트 간의 통신 중에 발생하는 에러 및 제어 정보를 제어하는 프로토콜

<오답 체크> **RARP**(Reverse Address Resolution Protocol) 프로토콜은 ARP의 반대로, MAC 주소를 IP 주소를 변환해주는 프로토콜이다.

7. 주기억 장치의 페이지 교체 기법에 대한 설명으로 가장 옳은 것은?

- ① FIFO(First In First Out)는 가장 오래된 페이지를 교체한다.
- ② MRU(Most Recently Used)는 최근에 적게 사용된 페이지를 교체한다.
- ③ LRU(Least Recently Used)는 가장 최근에 사용한 페이지를 교체한다.
- ④ LFU(Least Frequently Used)는 최근에 사용빈도가 가장 많은 페이지를 교체한다.

답 ①

① FIFO는 가장 먼저 들어온, 가장 오래 있었던 페이지를 교체한다.

<오답 체크> ② LFU에 대한 설명이다.

MRU는 가장 최근에 사용했던 페이지를 교체하는 기법으로, 특정 반복문의 경우 최근 사용했던 페이지보다 오래 전 사용되었던 페이지가 다음 차례에 사용될 가능성이 높는데, 이럴 때 사용한다.

③ MRU에 대한 설명이다.

LRU는 최근에 사용하지 않았던, 가장 오래 전에 사용했던 페이지를 교체하는 기법이다.

④ MFU에 대한 설명이다.

LFU는 최근에 가장 적게 사용한 페이지를 교체하는 기법이다.

◆ 페이지 교체 알고리즘

▶ **OPT**(OPTimal replacement, 최적 교체)

- 앞으로 가장 오랫동안 사용하지 않을 페이지를 교체하는 기법
- 각 페이지의 호출 순서와 참조 상황을 미리 예측해야 하므로 실현 가능성이 희박

▶ **FIFO**(First In First Out)

- 가장 먼저 들어와서 가장 오래 있었던 페이지를 교체하는 기법
- 벨레이디의 모순현상(Belady's Anomaly)이 발생
[일반적으로 페이지 프레임 수가 많으면 페이지 부재의 발생 횟수가 줄어들지만, 페이지 프레임 수를 증가시켰는데도 불구하고 페이지 부재 횟수가 더 많이 일어나는 현상]

▶ **LRU**(Least Recently Used)

- 최근에 가장 오랫동안 사용하지 않은 페이지를 교체하는 기법
- 계수기나 스택과 같은 별도의 하드웨어가 필요하며, 시간적인 오버헤드가 발생

▶ **LFU**(Least Frequently Used)

- 사용 빈도가 가장 적은 페이지를 교체하는 기법
- 프로그램 실행 초기에 많이 사용된 페이지가 그 후로 계속 페이지 프레임을 점유

▶ **NUR**(Not Used Recently)

- 최근에 사용하지 않은 페이지를 교체하는 기법
- 최근 사용 여부를 확인하기 위해서 각 페이지마다 참조 비트(Reference Bit)와 변형 비트(Modified Bit, Dirty Bit)를 사용
- 참조하지 않은 페이지를 우선 교체

▶ **MRU**(Most Recently Used)

- 가장 최근에 사용했던 페이지를 교체하는 기법(LRU의 반대)
- 특정한 반복 순환 구조 효율적

▶ **MFU**(Most Frequently Used)

- 사용 빈도가 가장 많은 페이지를 교체하는 기법
- 적게 사용된 페이지가 방금 들어온 것이고 아직 덜 사용되었으므로, 앞으로 사용될 확률이 높다는 계산에 근거

▶ **SCR**(Second Chance Replacement, 2차 기회 교체)

FIFO 기법의 단점을 보완하는 기법으로, 오랫동안 주기억장치에 있었지만 자주 사용되는 페이지의 교체를 방지하기 위한 기법이다.

각 페이지마다 참조 비트를 두어, 교체 대상이 되기 전에 참조 비트를 검사하여 1일 경우, 교체는 하지 않고 참조 비트만 0으로 바꾸어 한번의 기회를 더 부여한다. 그 다음에 참조되지 않아 참조 비트가 0으로 남아있는 상태에서 또 교체해야 할 상황이 오면 교체한다.

8. RAID(Redundant Array of Inexpensive Disks) 기술에 대한 설명으로 가장 옳지 않은 것은?

- ① RAID 1 레벨은 미러링(Mirroring)을 지원한다.
- ② RAID 3 레벨은 데이터를 블록 단위로 분산 저장하여 대용량의 읽기 중심 서버용으로 사용한다.
- ③ RAID 5 레벨은 고정적인 패리티 디스크 대신 패리티가 모든 디스크에 분산되어 저장되므로 병목 현상을 줄여준다.
- ④ RAID 6 레벨은 두 개의 패리티 디스크를 사용하므로 두 개의 디스크 장애 시에도 데이터의 복구가 가능하다.

답 ②

② RAID 3은 데이터를 바이트 단위로 저장하며, 패리티 코드를 이용해 오류를 체크한다. 블록 단위로 분산 저장하는 건 RAID 2이다.

◆ RAID(Redundant Array of Inexpensive/Independent Disk)

- ▷ RAID 0 스트라이핑(striping)
 - Striped disk array without fault tolerance
 - 단순 분산 저장 / 백업 가능 없음 / 오류 정정 없음
- ▷ RAID 1 미러링(mirroring)
 - Mirroring and duplexing
 - 완전 동일하게 복제 디스크 / 실제 저장용량은 절반
- ▷ RAID 2 해밍코드
 - Hamming code ECC
 - 오류 정정을 위해 해밍 코드 사용
- ▷ RAID 3 바이트 단위 패리티
 - Parallel transfer with parity
 - 패리티 전용 디스크 1개(바이트 단위)
- ▷ RAID 4 블록 단위 패리티
 - Independent data disks with shared parity disk
 - 패리티 전용 디스크 1개(블록 단위)
- ▷ RAID 5 분산 패리티
 - Independent data disks with distributed parity blocks
 - 패리티 블록을 각 디스크에 분산 저장 / 병목 현상 없음
- ▷ RAID 6 이중 패리티
 - Independent data disks with two independent distributed parity schemas
 - 패리티를 이 중으로 두 개의 디스크에 저장
- ▷ RAID 7
 - 컨트롤러로서 내장되어 있는 실시간 운영체계를 사용
 - 속도가 빠른 버스를 통한 캐시
- ▷ RAID 0+1보다 RAID 1+0이 안정성이 좋다.

9. 질의 최적화를 위한 질의문의 내부 형태 변화에 대한 규칙으로 가장 옳지 않은 것은?

- ① 선택(select) 연산은 교환적이다:

$$\sigma_{c1}(\sigma_{c2}(R)) \equiv \sigma_{c2}(\sigma_{c1}(R))$$
- ② 연속적인 프로젝트(project) 연산은 첫 번째 것을 실행하면 된다:

$$\Pi_{List1}(\Pi_{List2}(\dots(\Pi_{Listn}(R))\dots)) \equiv \Pi_{Listn}(R)$$
- ③ 합집합(U)과 관련된 프로젝트(project) 연산은 다음과 같이 변환된다: $\Pi(A \cup B) \equiv \Pi(A) \cup \Pi(B)$
- ④ 선택의 조건 c가 프로젝트 속성만 포함하고 있다면 교환적이다: $\sigma_c(\Pi(R)) \equiv \Pi(\sigma_c(R))$

답 ②

▷ 선택(SELECT, σ)는 조건을 만족하는 튜플을 구하는 연산 프로젝트(PROJECT, Π)는 조건을 만족하는 속성을 구하는 연산

② 연속적인 프로젝트 연산에서는 가장 나중의 것을 실행하면 된다.
$$\Pi_{List1}(\Pi_{List2}(\dots(\Pi_{Listn}(R))\dots)) \equiv \Pi_{List1}(R)$$
전체 데이터에서 성별, 나이, 연봉 속성을 추출한 뒤
-> 그 중에서 성별, 연봉 속성을 추출한 뒤
-> 그 중에서 성별 속성을 추출하는 것은
전체 데이터에서 성별 속성만 추출하는 것과 같다.

<오답 체크> ① 선택 연산은 교환법칙이 성립한다.

$$\sigma_{<조건1>}(\sigma_{<조건2>}(R)) = \sigma_{<조건2>}(\sigma_{<조건1>}(R))$$
월급이 300 이상인 사원들을 추출한 후 여자 사원을 추출하는 것이나, 여자 사원을 추출한 후 월급이 300 이상인 사원을 추출하는 것이나 같다.
반면에 프로젝트 연산은 교환법칙이 성립하지 않는다.

② 부서와 연봉 속성을 함께(합집합) 추출하는 것이나 부서 속성을 추출하고 연봉 속성을 추출하여 두 릴레이션을 합치는 것(합집합)은 같다.

④ 부서 속성을 추출한 후 회계부서 사원을 추출하는 것이나, 회계부서 사원을 추출한 후 부서 속성만 추출하는 것은 같다.
선택의 조건 C가 프로젝트의 속성만 포함하고 있다는 말은, 프로젝트 조건으로 부서 속성을 추출하는 것으로 정해놓으면, 선택 조건으로는 성별이나 연봉 등 다른 속성 조건으로 집어넣지 말라는 의미이다.
연봉이 5천 이상인 사원을 추출한 후 부서 속성을 추출하는 것은 가능하나, 부서 속성을 추출한 상태에서는 연봉 속성이 없기 때문에 연봉이 5천 이상인 사원을 추출할 수 없게 된다.

16. <보기> 회로의 종류를 바르게 연결한 것은?

< 보 기 >	
ㄱ. 3개의 입력 중에서 적어도 2개의 입력이 1이면 출력이 1이 되는 회로	
ㄴ. 설정된 값이 표시되었을 때, 경고음을 울리는 카운터	

- | | |
|----------|--------|
| ㄱ | ㄴ |
| ① 조합논리회로 | 조합논리회로 |
| ② 조합논리회로 | 순차논리회로 |
| ③ 순차논리회로 | 조합논리회로 |
| ④ 순차논리회로 | 순차논리회로 |

답 ②

- ㄱ. 입력값에 따라 출력값이 결정되는 회로이므로 조합 논리회로에 해당한다.
- ㄴ. 이미 어떠한 설정된 값으로 들어있는 상태에서 현재 표시되는 값과 비교하여 출력을 따지는 회로이기 때문에 순차 논리회로에 해당한다.

- ▶ 조합 논리회로(combinational logic circuit)
출력값이 입력값에 의해서만 결정
가산기, 비교기, 연산기, 디코더, 인코더, 멀티플렉서, 디멀티플렉서, 코드 변환기 등
- ▶ 순차 논리회로(sequential logic circuit)
출력값이 입력값과 이전 상태의 논리값에 의해 결정
플립플롭, 카운터, 레지스터, RAM, CPU 등

17. CISC(Complex Instruction Set Computer)에 대한 설명으로 가장 옳은 것은?

- ① 고정 길이의 명령어 형식을 가진다.
- ② 명령어의 길이가 짧다.
- ③ 다양한 어드레싱 모드를 사용한다.
- ④ 하나의 명령으로 복잡한 명령을 수행할 수 없어 복잡한 하드웨어가 필요하다.

답 ③

- ③ 다양한 명령어 형식을 사용하는 건 CISC이다. 다양한 어드레싱 모드를 사용한다는 것은 명령어 주소 형식을 직접 주소 방식, 간접 주소 방식, 상대적 주소 방식 등의 여러 가지 형식으로 사용한다는 것이다.

<오답 체크> ①② 고정 길이의 상대적으로 짧은 명령어를 사용하는 것은 RISC 방식이다.

- ④ CISC 방식은 하나의 명령어로 다양하고 복잡한 명령을 수행할 수 있으며, 따라서 연산을 위한 하드웨어(회로도) 역시 복잡하다. 반면, RISC 방식은 명령어가 짧고 단순하여 복잡한 명령을 수행하기 부적합하며, 하드웨어(회로도)를 논리 순서대로 설계하여 처리 속도를 향상시켰다.

◆ CISC (Complex Instruction Set Computer)

- 다양한 길이의 명령어
- 회로가 복잡하고 동시에 여러 개의 명령어 처리가 어려움
- 컴파일러 작성이 쉽고 호환성이 좋음
- 마이크로 프로그래밍 방식

◆ RISC (Reduced Instruction Set Computer)

- 고정된 길이의 명령어
- 명령어 처리 속도가 빠르며 동시에 여러 명령어 처리 가능
- 파이프라인 성능에 최적화(파이프라인 해저드 주의)
- 컴파일러 최적화 과정이 복잡하고 코드 효율성이 낮음
- 많은 수의 범용 레지스터
- 하드와이어드(Hardwired) 방식

18. 퀵 정렬에 대한 설명으로 가장 옳지 않은 것은?

- ① 퀵 정렬은 분할 정복(divide and conquer) 방식으로 동작한다.
- ② 퀵 정렬의 구현은 흔히 재귀 함수 호출을 포함한다.
- ③ n개의 데이터에 대한 퀵 정렬의 평균 수행 시간은 $O(\log n)$ 이다.
- ④ C.A.R. Hoare가 고안한 정렬 방식이다.

답 ③

③ 퀵 정렬의 평균 시간 복잡도는 $O(n \log n)$ 이며, 최악의 경우 $O(n^2)$ 이다.

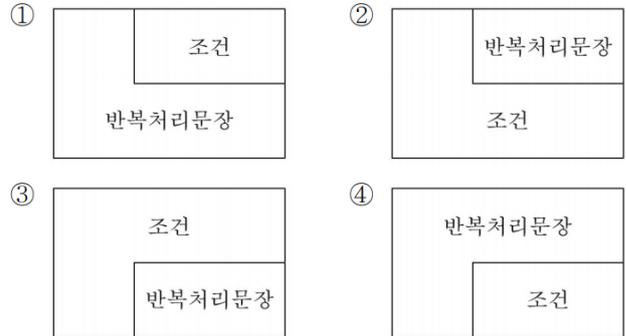
<오답 체크> ① 퀵 정렬 알고리즘은 분할 정복 방법을 통해 리스트를 정렬한다.

② 퀵 정렬 알고리즘을 구현할 때 보통 재귀적 방법을 이용하여 구현한다.

◆ 정렬 알고리즘 시간복잡도(Time complexity)

알고리즘	최선	평균	최악
삽입 정렬	$O(n)$	$O(n^2)$	$O(n^2)$
선택 정렬	$O(n^2)$	$O(n^2)$	$O(n^2)$
버블 정렬	$O(n^2)$	$O(n^2)$	$O(n^2)$
셸 정렬	$O(n)$	$O(n^{1.5})$	$O(n^2)$
퀵 정렬	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n^2)$
힙 정렬	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n \log_2 n)$
합병 정렬	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n \log_2 n)$
기수 정렬	$O(dn)$	$O(dn)$	$O(dn)$

19. 나시-슈나이더만(N-S) 차트의 반복(While) 구조에 대한 표현으로 가장 옳은 것은?



답 ③

▶ 나시-슈나이더만(N-S) 차트

구조적 프로그램의 순차, 선택, 반복의 구조를 사각형으로 도식화하여 알고리즘을 논리적으로 표현한 기법

③ while문은 조건을 먼저 확인한 후 해당 반복 문장을 수행하기 때문에, ③번과 같이 조건이 위로 반복문이 아래로 간 형태로 표현된다.

<오답 체크> ② do while문의 경우는 먼저 반복문을 한 번 먼저 수행한 후 조건을 따져 계속 반복문을 수행한다. 따라서 ②번처럼 반복문이 위로 조건문이 아래로 간 형태로 표현된다.

