

2022. 06. 18 9급 지방직 공무원 공개경쟁 채용시험 총평

컴퓨터일반 : 최희준 선생님

한
줄
평

▶ 출제 난이도와 경향



새로운 문제들이 다수(5개)가 출제되었습니다.

새롭게 출제된 문제를 제외한 나머지 문제들은 난이도가 낮아 쉽게 풀 수 있는 문제가 출제되었습니다.

새롭게 출제된 문제 유형들 중에는 정말 어려운 문제가 2개 출제되었고,

나머지 3개는 공부했다면 쉽게 풀 수 있는 문제였습니다.

예전부터 중요하게 다루던 문제들은 계속 출제될 것으로 보이며, 평균 3개 정도의 새로운 문제나 어려운 문제를 출제할 것으로 보입니다.

▶ 출제 유형분석 (주제별, 문제유형별 출제문항 수)

문제번호	출제과목	내용	출제정보(책)	난이도
1번	프로그래밍어	알고리즘 조건	[출인원 7.03.17] 알고리즘(Algorithm), [출인원 7.03.18] 순서도(Flow Chart)	하
2번	데이터베이스	빅데이터 특성	[출인원 6.06.09] 빅데이터(Big Data)	하
3번	자료구조	삽입 정렬 알고리즘	[출인원 5.04.04] 삽입 정렬(Insertion Sort) - 내부 정렬	하
4번	소프트웨어공학	화이트박스 테스트	[출인원 3.07.02] 검사 방법	하
5번	자료구조	진법 변환	[출인원 5.01.03] 10진수를 2, 8, 16진수로 변환	하
6번	운영체제	교착상태 해결방안 은행원 알고리즘	[출인원 1.02.21] 교착상태의 해결방안	하
7번	정보통신개론	OSI 계층의 기능, 장비 구분	[출인원 2.07.05] OSI 7 계층의 기능	하
8번	전자계산기구조	이미지 표현에서 RGB, CMYK 구분	새롭게 출제	하
9번	데이터베이스	지연 갱신 회복의 redo, undo 이해	새롭게 출제	상
10번	소프트웨어공학	CMMI 성숙 단계 구분 문제	새롭게 출제	중
11번	전자계산기구조	JK 플립플롭 상태 구분문제	[출인원 4.05.09] 플립플롭(Flip-Flop) = 1 비트 기억소자	하
12번	데이터베이스	INSERT 문법 문제	[출인원 6.05.12] INSERT 명령어	하
13번	정보통신개론	패킷 통신 특징	[출인원 2.06.03] 메시지 교환 방식과 패킷 교환 방식	하
14번	운영체제	인터럽트 틀린 설명 찾기	[출인원 1.02.04] 인터럽트 처리(Interrupt Processing)	하
15번	프로그래밍어	C언어 실행 결과, 배열, 반복문 for	[출인원 7.05.29] 배열의 사용	하
16번	정보통신개론	TCP의 임계값과 혼잡 윈도우값 찾기	새롭게 출제	상
17번	운영체제	가변 분할(동적 분할) 할당 기법	[출인원 1.03.03] 주기억장치-다중프로그래밍	하
18번	운영체제	UMA와 NUMA의 구분 문제	새롭게 출제	중
19번	프로그래밍어	C언어 실행 결과, 외부 내부 변수 차이	[출인원 7.05.32] 기억 클래스(Storage Class)	중
20번	운영체제	프로세스 스케줄링 SJF 평균 반환시간 계산	[출인원 1.02.10] SJF(Short Job First) - 비선점형	중

총
평

▶ 주목해야 하는 이슈 문항

(변별력을 가르는 고난도 문항 or 기존에는 출제되지 않았던 신유형 문항)

변별력을 가르는 새로운 유형에 문제와 2개의 어려운 문제가 출제되었습니다. 나머지 문제들은 자주 출제되는 중요한 문제들이 출제되었으며 난이도는 매우 낮았습니다.

특히, 새롭게 출제된 소프트웨어 공학에서의 CMMI 라는 소프트웨어의 개선 모델과

정보통신개론의 TCP의 임계값과 혼잡 윈도우값 찾기 문제는 앞으로도 계속 다뤄질 것으로 보입니다.

그리고, C언어의 결과 찾기는 문제는 기존에 출제된 내용들과 비교해서 매우 쉬었다고 할 수 있습니다.

▶ 2023 대비 학습법 (다음 시험을 준비하는 학생들이 주목해야 할 학습방향)

전통적으로 중요하게 다루는 문제는 항상 80%이상 출제되고 있습니다.

기존의 이론(우리책)은 완벽하게 습득하시어 시험장에서 공부하신 내용이 헛갈리지 않도록 하는 것이 중요합니다.

나머지 어려운 문제나 최신에 다뤘던 새로운 문제들은 다양한 종목들에서 출제된 문제들을 통해 어느 정도의 경향을 파악할 수 있습니다.

특히 "컴퓨터일반 기출문제 끝장내기"에 있는 계리직 기출문제, 동형 모의고사 문제 등도 빠짐없이 이해하시기 바랍니다.

문 1. 컴퓨터 알고리즘의 조건에 대한 설명으로 옳지 않은 것은?

- ① 각 명령어의 의미는 모호하지 않고 명확해야 한다.
- ② 알고리즘 단계들에는 순서가 정해져 있지 않다.
- ③ 한정된 수의 단계 후에는 반드시 종료되어야 한다.
- ④ 각 명령어들은 실행 가능한 연산이어야 한다.

[정답] ②

[해설]

컴퓨터 프로그램은 특정 목적을 위해 단계적으로 CPU에게 명령을 주어 처리하게 해야 하는데, 그 단계가 여러 가지의 명령이 복합적이면서 체계적이고 합리적으로 이루어져야 하기 때문에 알고리즘이라고 하는 것이다.

[출제정보] [출인원 7.03.17] 알고리즘(Algorithm), [출인원 7.03.18] 순서도(Flow Chart)

[난이도] 하

문 2. 다음에서 설명하는 빅데이터의 3대 특징으로 옳지 않은 것은?

빅데이터는 대용량의 데이터 집합으로부터 가치 있는 정보를 효율적으로 추출하고 결과를 분석하는 기술이다.

- ① 센싱 기술 등을 활용하여 사물과 주위 환경으로부터 정보 획득(sensor)
- ② 방대한 양의 데이터 처리(volume)
- ③ 정형 데이터와 비정형 데이터 등 다양한 유형의 데이터로 구성(variety)
- ④ 실시간으로 생산되며 빠른 속도로 수집 및 분석(velocity)

[정답] ①

[해설]

빅데이터의 특성을 나타내는 3V는 규모(Volume), 속도(Velocity), 형태의 다양성(Variety)를 의미한다.

[출제정보] [출인원 6.06.09] 빅데이터(Big Data)

[난이도] 하

문 3. 다음 자료를 오름차순으로 삽입 정렬(insertion sort)하는 과정에서 나올 수 없는 경우는?

3 1 4 2 9 5

- ① 1 3 4 2 9 5
- ② 1 2 3 4 9 5
- ③ 3 1 5 2 4 9
- ④ 1 2 3 4 5 9

[정답] ③

[해설]

	1	2	3	4	5	6
정렬 대상	3	1	4	2	9	5

Pass 1	[1]	[3]	4	2	9	5
Pass 2	[1]	[3]	[4]	2	9	5
Pass 3	1	2	3	[4]	9	5
Pass 4	1	2	3	4	[9]	5
Pass 5	1	2	3	4	5	[9]

정렬 결과	1	2	3	4	5	9
-------	---	---	---	---	---	---

[출제정보] [출인원 5.04.04] 삽입 정렬(Insertion Sort) - 내부 정렬
[난이도] 하

문 4. 소프트웨어의 화이트박스 테스트에 대한 설명으로 옳지 않은 것은?

- ① 글래스 박스(Glass-box) 테스트라고 부른다.
- ② 소프트웨어의 내부 경로에 대한 지식을 보지 않고 테스트 대상의 기능이나 성능을 테스트하는 기술이다.
- ③ 문장 커버리지, 분기 커버리지, 조건 커버리지 등의 검증 기준이 있다.
- ④ 모듈의 논리적인 구조를 체계적으로 점검하기 때문에 구조적 테스트라고도 한다.

[정답] ②

[해설]

②번의 설명은 기능이나 성능을 테스트하는 블랙박스 테스트의 설명이다. 화이트박스 테스트는 소프트웨어의 내부 경로에 대한 지식을 보면서 테스트 한다.

[출제정보] [출인원 3.07.02] 검사 방법
[난이도] 하

문 5. 16진수 210을 8진수로 변환한 것은?

- ① 1020
- ② 2100
- ③ 10210
- ④ 20100

[정답] ①

[해설]

2진수	0	0	1	0	0	0	0	1	0	0	0	0
16진수	2				1				D			

2진수	0	0	1	0	0	0	0	1	0	0	0	0
	■			■			■			■		
8진수	1			0			2			0		

[출제정보] [출인원 5.01.03] 10진수를 2, 8, 16진수로 변환
[난이도] 하

문 6. 은행원 알고리즘(banker's algorithm)이 교착상태를 해결하는 방법은?

- ① 예방
- ② 회피
- ③ 검출
- ④ 회복

[정답] ②

[해설]

- ① 예방 : 교착상태 발생의 필요 충분 조건 4가지를 부정
- ② 회피 : 은행원 알고리즘
- ③ 검출(발견) : 인접행렬로 표현
- ④ 회복 : 희생양 결정

[출제정보] [출인원 1.02.21] 교착상태의 해결방안

[난이도] 하

문 7. 다음 OSI 7계층 중 물리 계층에 해당하는 장치를 모두 고른 것은?

- ㄱ. 리피터(Repeater)
- ㄴ. 더미허브(Dummy Hub)
- ㄷ. 라우터(Router)
- ㄹ. 게이트웨이(Gateway)
- ㅁ. 브릿지(Bridge)

- ① ㄱ, ㄴ
- ② ㄱ, ㄷ
- ③ ㄴ, ㄹ
- ④ ㄹ, ㅁ

[정답] ①

[해설]

물리계층 : 케이블, 리피터, 더미 허브

데이터링크 계층 : 브릿지

네트워크 계층 : 라우터

OSI 전계층 : 게이트웨이

[출제정보] [출인원 2.07.05] OSI 7 계층의 기능

[난이도] 하

문 8. 이미지 표현을 위한 RGB 방식과 CMYK 방식에 대한 설명으로 옳은 것은?

- ① CMYK 방식은 가산 혼합 모델로 빛이 하나도 없을 때 검은색을 표현한다.
- ② CMYK 방식에서 C는 Cyan을 의미한다.
- ③ RGB 방식은 주로 컬러 프린터, 인쇄, 페인팅 등에 적용된다.
- ④ RGB 방식에서 B는 Black을 의미한다.

[정답] ②

[해설]

■ RGB와 CMYK

• RGB : Red, Green, Blue를 의미하며 빛의 혼합 방식으로 3가지 색이 혼합되면 색이 점점 밝아져 흰색(White : 코드 #FFFFFF)이 된다. 이러한 방식을 가산 혼합이라고 하며 컴퓨터, 스마트폰 액정, TV에서 주로 사용한다.

• CMYK : Cyan(청록), Magenta(자주), Yellow(노랑), black(검정)를 의미하며 색(물감)의 혼합 방식으로 3가지 색(CMY)이 혼합되면 색이 점점 어두어져 검은색(black, #000000)이 된다. 이러한 방식을 감산 혼합이라고 하며, 컬러 프린터, 인쇄, 페인팅 등에 사용된다.

틀린 설명으로 올바르게 고치면 다음과 같다.

- ① CMYK 방식은 감산 혼합 모델로 색을 모두 혼합하면 검은색이 된다.
- ③ RGB 방식은 주로 컴퓨터, 스마트폰 액정, TV에서 주로 사용한다.
- ④ RGB 방식에서 B는 Blue를 의미한다.

[출제정보] [출원원 4.04.12] 디스플레이 어댑터와 모니터 - 새롭게 출제

[난이도] 하

문 9. 다음은 A 계좌에서 B 계좌로 3,500원을 이체하는 계좌 이체 트랜잭션 T1과, C 계좌에서 D 계좌로 5,200원을 이체하는 계좌 이체 트랜잭션 T2가 순차적으로 수행되면서 기록된 로그파일 내용이다. (가)의 시점에서 장애가 발생했을 경우 지연 갱신 회복 기법을 적용했을 때 트랜잭션에 대한 회복조치로 옳은 것은?

```

1: <T1, start>
2: <T1, A, 7800>
3: <T1, B, 3500>
4: <T1, commit>
5: <T2, start>
6: <T2, C, 9820>
   (가)
7: <T2, D, 5200>
8: <T2, commit>

```

- ① T1, T2 트랜잭션 모두 별다른 조치를 수행하지 않는다.
- ② T1 트랜잭션의 로그 내용을 무시하고 버린다.
- ③ T1 트랜잭션에는 별다른 회복조치를 하지 않지만, T2 트랜잭션에는 redo(T2) 연산을 실행한다.
- ④ T2 트랜잭션에는 별다른 회복조치를 하지 않지만, T1 트랜잭션에는 redo(T1) 연산을 실행한다.

[정답] ④

[해설]

T1 : 지연 갱신 회복 기법을 적용하면 장애가 발생하지 전에 완료(Commit) 되었으므로 Redo를 수행한다.

T2 : 지연 갱신 회복 기법을 적용하면 트랜잭션 중간에 장애가 발생하였으므로 이미 진행된 작업은 로그 파일에만 존재하므로 로그 파일만 제거하면 된다. 데이터베이스에는 적용되지 않았으므로 별다른 조치를 하지 않는다. (No Undo)

T1 : 즉시 갱신 회복 기법을 적용하면 장애가 발생하지 전에 완료(Commit) 되었으므로 Redo를 수행한다.

T2 : 즉시 갱신 회복 기법을 적용하면 트랜잭션 중간에 장애가 발생하였으므로 이미 적용되었던 데이터베이스를 Undo하여 회복한다.

■ 회복 기법

(1) 그림자 페이징(Shadow Paging) 회복

- 로그(Log) 파일을 이용하지 않고, 디스크에 백업받아 놓은 파일로 대체하는 방법이다.
- 현재 데이터베이스의 테이블에 변환된 연산들을 그대로 디스크에서도 동일하게 그림자처럼 저장한다.
- 장애가 발생하여 회복이 필요할 경우에 그림자처럼 저장되었던 그림자 파일로 대체한다.
- 연산 작업이 큰 경우에는 여러 개의 페이지 블록으로 구분하여 처리하므로 그림자 페이징 회복이라고 한다.
- 페이지 블록이 변경 시 현재 페이지 블록만 변경하고 그림자 페이지는 이전 값을 유지한다.

(2) 즉시 갱신(Immediate update) 회복

- 로그(Log) 파일을 이용하는 방법으로 처리된 작업들을 데이터베이스에 즉시 적용시킨다.
- 특정 데이터 값이 매번 변경되더라도 그때마다 계속적으로 데이터베이스에 적용한다.

- 중간에 장애가 발생한 경우에는 장애시점부터 Undo를 시행한다.
- 로그에 기억된 값 관리, Undo를 수행하므로 복잡하다.

(3) 지연 갱신(Deferred update) 회복

- 로그(Log) 파일을 이용하는 방법으로 작업이 완전히 종료될 때까지 지연시켰다가 데이터베이스에 적용시킨다.
- 연산으로 인하여 특정 데이터 값이 매번 변경되더라도 로그 파일에만 저장한다.
- 모든 작업이 완료되었을 경우에 데이터베이스에 적용시킨다.
- 중간에 장애가 발생할 경우에는 로그 파일만 제거하면 된다. (NoUndo)
- 로그에 기억된 값만 관리하므로 단순하다.

(4) 검사점(Checkpoint) 회복

- 검사점 이전까지는 아무런 문제없이 진행된 트랜잭션이므로 회복에서 제외한다.
- 검사점 존재하는 트랜잭션이 장애가 발생하기 전에 완료되었다면 검사점 이후에 발생한 연산들은 Redo를 수행한다.
- 검사점 존재하는 트랜잭션이 완료가 되지 않고 장애가 발생하였다면 검사점 이후에 발생한 연산들은 Undo를 수행한다.
- 쓸데없는 부가적인 노력을 방지하는데 목적이 있다.

[출제정보] [출원원 6.06.03] 트랜잭션 상태 - 새롭게 출제

[난이도] 상

문 10. 다음에 해당하는 CMMI(Capability Maturity Model Integration) 모델의 성숙 단계로 옳은 것은? (단, 하위 성숙 단계는 모두 만족한 것으로 가정한다)

- | | |
|------------------|------------------|
| ○ 요구사항 개발 | ○ 조직 차원의 프로세스 정립 |
| ○ 기술적 솔루션 | ○ 조직 차원의 교육훈련 |
| ○ 제품 통합 | ○ 통합 프로젝트 관리 |
| ○ 검증 | ○ 위험관리 |
| ○ 확인 | ○ 의사 결정 분석 및 해결 |
| ○ 조직 차원의 프로세스 개선 | |

- ① 2단계
② 3단계
③ 4단계
④ 5단계

[정답] ②

[해설]

■ CMMI의 역량 성숙도

(1) CMMI의 역량 성숙도

- 프로세스 프레임워크의 성숙도 향상을 위한 모델이다.
- 소프트웨어 개발 과정에서의 비용, 품질, 일정 등 모든 것을 충족시키기 위한 조직의 프로세스 개선 모델이다.
- 특정 성숙도 단계가 되기 위한 최소한의 기준 제시와 반드시 수행해야 할 활동들의 집합이다.
- CMMI의 역량 성숙도는 1단계~5단계로 나뉘어져 있다. 1단계는 매우 미숙하고 혼돈된 프로세스이며, 5단계는 최적화된 가장 성숙한 최고 수준의 프로세스이다.

(2) CMMI의 역량 성숙도 5단계(Level)

- 1단계(Initial) : 개인의 역량에 따라 프로젝트의 성공과 실패가 좌우된다. 정상적인 프로세스에 의해서 해결할 수 없는 경우 임시로 문제를 해결하기 위해 만든 후 해체하는 매우 미숙하고 혼돈된 조직이다. (Ad-hoc 프로세스 조직)

- 적용 프로세스가 없음

- 2단계(Managed) : 기본적인 프로세스 구축에 의해 프로젝트가 관리되고 있는 조직으로 일정이나 비용과 같은 관리 프로세스 중심이다. (기존 유사 성공사례 반복 사용)

- 요구사항 관리(Requirement Management)
- 프로젝트 계획(Project Planning)
- 프로젝트 감시 및 제어(Project Monitoring & Control)
- 공급자 합의 관리(Supplier Agreement Management)
- 측정과 분석(Measurement & Analysis)
- 프로세스와 제품 품질 보증(Process & Product Quality Assurance)
- 형상관리(Configuration Management)

- 3단계(Defined) : 조직을 위한 표준 프로세스가 존재하며 모든 프로젝트는 조직의 표준 프로세스를 이용하여 상황에 맞게 조정하여 승인받아 사용한다. (표준 프로세스로 프로젝트를 통제)

- 요구사항 개발(Requirement Development)
- 기술적 솔루션(Technical Solution)
- 제품 통합(Product Integration)
- 검증(Verification)
- 조직 차원의 프로세스 개선(Organization Process Focus)
- 조직 차원의 프로세스 정립(Organization Process Definition)
- 조직 차원의 교육 훈련(Organization Training)
- 통합 프로젝트 관리(Integrated Project Management)
- 통합 공급자 관리(Integrated Supplier Management)
- 위험관리(Risk Management)
- 의사 결정 분석 및 해결(Decision Analysis & Revolution)
- 통합을 위한 조직적 환경(Organizational Environment for Integration)
- 통합된 팀 구성(Integrated Teaming)

- 4단계(Quantitatively Managed) : 소프트웨어 품질에 대한 정량적인 측정이 가능해진다. 데이터베이스를 구축하여 프로젝트에서 측정된 결과를 일괄적으로 수집하고 분석하여 품질평가를 한다. (관리와 통제가 되며 성과 예측이 가능한 조직)

- 조직 차원의 프로세스 성과(Organizational Process Performance)
- 정량적인 프로젝트 관리(Quantitative Project Management)

- 5단계(Optimizing) : 조직적으로 최적화된 프로세스를 적용하고 피드백(Feedback)을 통하여 개선한다. (지속적인 개선활동이 정착, 최적의 관리로 프로젝트가 수행)

- 조직적 혁신과 배치(Organization Innovation & Deployment)
- 분석과 해결(Casual Analysis & Revolution)

[출제정보] 출제 정보 없음 - 새롭게 출제
[난이도] 중

문 11. 다음은 정논리를 사용하는 JK 플립플롭의 진리표이다. (가)~(라)에 들어갈 내용으로 옳은 것은? (단, Q'은 Q의 반댓값을 의미한다)

CP	J	K	다음상태 Q
↑	0	0	(가)
↑	0	1	(나)
↑	1	0	(다)
↑	1	1	(라)

	(가)	(나)	(다)	(라)
①	Q	1	0	Q'
②	Q'	1	0	Q
③	Q	0	1	Q'
④	Q'	0	1	Q

[정답] ③
[해설]

J	K	현재 Q(t)	다음 Q(t+1)	설명
0	0	0	0	상태변화 없음(불변)
		1	1	
0	1	0	0	0인 상태로 변경(리셋)
		1	0	
1	0	0	1	1인 상태로 변경(세트)
		1	1	
1	1	0	1	토글(Toggle) (반전)
		1	0	

[출제정보] [출인원 4.05.09] 플립플롭(Flip-Flop) = 1비트 기억소자
[난이도] 하

문 12. 다음 SQL(Structured Query Language)문으로 생성한 테이블에 내용을 삽입할 때 올바르게 동작하지 않는 SQL 문장은?

```
CREATE TABLE Book (ISBN CHAR(17) PRIMARY KEY,
TITLE VARCHAR(30) NOT NULL, PRICE INT NOT NULL,
PUBDATE DATE, AUTHOR VARCHAR(30));
```

- ① INSERT INTO Book (ISBN, TITLE, PRICE, AUTHOR) VALUES ('978-89-8914-892-1', '데이터베이스 개론', 20000, '홍길동');
- ② INSERT INTO Book VALUES ('978-89-8914-892-2', '데이터베이스 개론', 20000, '2022-06-18', '홍길동');
- ③ INSERT INTO Book (ISBN, TITLE, PRICE) VALUES ('978-89-8914-892-3', '데이터베이스 개론', 20000);
- ④ INSERT INTO Book (ISBN, TITLE, AUTHOR) VALUES ('978-89-8914-892-4', '데이터베이스 개론', '홍길동');

[정답] ④

[해설]

- ①번은 생성된 테이블의 필드를 선언(ISBN, TITLE, PRICE, AUTHOR)하고 VALUE를 순서와 타입에 맞게 입력하였다.
- ②번은 생성된 테이블의 필드 선언을 생략하고 VALUE를 순서와 타입에 맞게 입력하였다. 필드 생성된 순서대로 VALUE를 입력하면 필드 선언은 생략할 수 있다.
- ③번은 AUTHER이 입력되지 않아도 올바르게 동작한다. 테이블을 생성할 때 AUTHER는 아무런 조치를 취하지 않았으므로(NOT NULL과 같은 정의) 오류가 발생하지 않는다.
- ④번에서 PRICE는 입력되지 않아서 올바르게 동작하지 않는다. 테이블을 생성할 때 PRICE는 정수형(INT)이며 NULL이 될 수 없다고(NOT NULL) 정의했기 때문에 PRICE는 반드시 필드선언과 VALUE가 필요하다.

[출제정보] [출인원 6.05.12] INSERT 명령어

[난이도] 하

문 13. 패킷 교환 네트워크에 대한 설명으로 옳지 않은 것은?

- ① 패킷 크기는 옥텟(Octet) 단위로 사용한다.
- ② 네트워크로 전송되는 모든 데이터는 송-수신지 정보를 포함하는 패킷들로 구성된다.
- ③ 패킷 교환 방식은 접속 방식에 따라 데이터그램 방식과 가상회선 방식이 있다.
- ④ 패킷 교환 네트워크에서는 동시에 2쌍 이상의 통신이 불가능하다.

[정답] ④

[해설]

패킷 교환은 송수신을 동시에 할 수 있는 전이중 통신이다. 따라서 패킷 교환 네트워크에서는 동시에 2쌍 이상의 통신이 가능하다. 패킷의 크기는 바이트(옥텟(Octet)) 단위로 사용된다.

[출제정보] [출인원 2.06.03] 메시지 교환 방식과 패킷 교환 방식

[난이도] 하

문 14. 인터럽트에 대한 설명으로 옳지 않은 것은?

- ① 내부 인터럽트가 발생하면 컴퓨터는 더 이상 프로그램을 실행할 수 없다.
- ② 프로세서는 인터럽트 요구가 있으면 현재 수행 중인 프로그램의 주소 값을 스택이나 메모리의 0번지와 같은 특정 장소에 저장한다.
- ③ 신속하고 효율적인 인터럽트 처리를 위하여 컴퓨터는 항상 인터럽트 요청을 승인하도록 구성된다.
- ④ 인터럽트 핸들러 또는 인터럽트 서비스 루틴은 인터럽트 소스가 요청한 작업에 대한 프로그램으로 기억장치에 적재되어야 한다.

[정답] ③

[해설]

인터럽트 요청은 상황에 따라 언제든지 요청이 취소될 수 있다.

[출제정보] [출원인 1.02.04] 인터럽트 처리(Interrupt Processing)

[난이도] 하

문 15. 다음 C 프로그램을 실행하면서 사용자가 1, 2, 3, 4를 차례대로 입력했을 때, 출력 결과는?

```
01 #include <stdio.h>
02 int main()
03 {
04     int ary[4];
05     int sum = 0;
06     int i;
07
08     for (i = 0; i < 4; i++) {
09         printf("%d번 째 값을 입력하시오 : ", i + 1);
10         scanf("%d", &ary[i]);
11     }
12
13     for (i = 3; i > 0; i--)
14         sum += ary[i];
15
16     printf("%d \n", sum);
17     return 0;
18 }
```

- ① 3
- ② 6
- ③ 9
- ④ 10

[정답] ③

[해설]

04 : 정수형 배열 4개를 확보하였다

ary[0]	ary[1]	ary[2]	ary[3]

05: 14 라인의 sum+= ary[i]에서 사용할 누적변수를 sum을 선언하고 0으로 초기화하였다.

06 : 08라인이나 14라인에서 사용할 반복변수 i를 선언하였다.

08 : 반복변수 혹은 배열의 첨자변수로 사용할 변수 i를 0부터 3까지 1씩 증가하면서 반복구간 09라인~11라인까지를 반복한다.

09 : i가 0일 때 “1번째 값을 입력하세요 :” 출력,
 i가 1일 때 “2번째 값을 입력하세요 :” 출력,
 i가 2일 때 “3번째 값을 입력하세요 :” 출력,
 i가 3일 때 “4번째 값을 입력하세요 :” 출력한다.

10 : i가 0일 때 ary[0]에 입력한 값을 기억,
 i가 1일 때 ary[1]에 입력한 값을 기억,
 i가 2일 때 ary[2]에 입력한 값을 기억,
 i가 3일 때 ary[3]에 입력한 값을 기억시킨다.

scanf() 함수는 입력될 변수의 번지를 지정해야하므로 반드시 변수명 앞에 &를 붙여
 사용하거나 배열명만을 사용해야한다.

&ary[0] == ary

&ary[1] == ary+1

&ary[2] == ary+2

&ary[3] == ary+3

문제에서 1, 2, 3, 4를 차례대로 입력하므로

1번째 값을 입력하세요 : 1

2번째 값을 입력하세요 : 2

3번째 값을 입력하세요 : 3

4번째 값을 입력하세요 : 4

형태로 화면에 출력되고, 입력하면서 다음과 같이 기억된다.

1	2	3	4
ary[0]	ary[1]	ary[2]	ary[3]

13 : 반복변수 혹은 배열의 첨자변수로 사용할 변수 i를 3부터 1까지 1씩 감소하면서 반복구간 11라인을 반복한다.
 최종비교치가 “i > 0”이므로 i가 0일 때는 수행하지 않는다.

14 : 반복변수 i가 3, 2, 1까지만 수행되므로

sum += ary[3];

sum += ary[2];

sum += ary[1]; 만 수행한다.

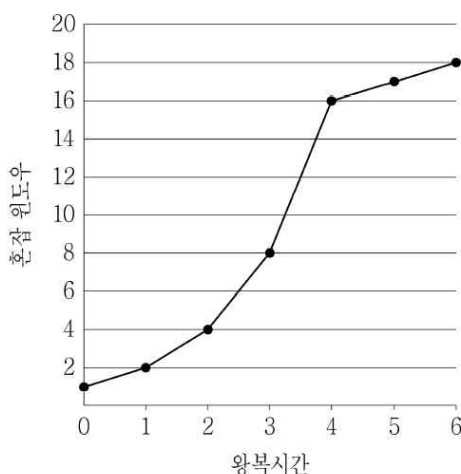
따라서 sum = ary[3] + ary[2] + ary[1] = 4 + 3 + 2 = 9, sum에는 9가 기억된다.

16 : sum을 정수형으로 출력하므로 9가 출력된다.

[출제정보] [출인원 7.05.29] 배열의 사용

[난이도] 하

문 16. 그림은 TCP Tahoe에서 데이터 전송에 따른 혼잡 윈도우(cwnd, 단위 : MSS)의 크기 변화를 나타낸다. 혼잡 윈도우값이 18일 때의 전송에서 Time-out이 발생했을 때, 느린 출발(slow-start) 임꺽값과 혼잡 윈도우값 변화로 옳은 것은?



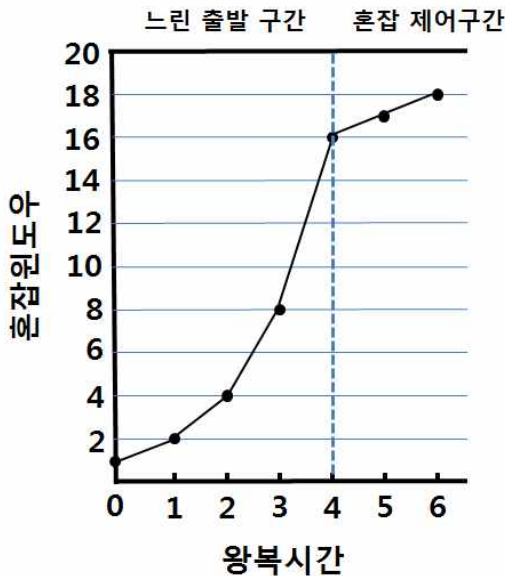
- ① 임계값은 변하지 않고, 혼잡 윈도우값은 1로 감소한다.
- ② 임계값이 9가 되고, 혼잡 윈도우값은 1로 감소한다.
- ③ 임계값이 9가 되고, 혼잡 윈도우값은 현재 값의 반으로 감소한다.
- ④ 임계값은 변하지 않고, 혼잡 윈도우값은 현재 값의 반으로 감소한다.

[정답] ②

[해설]

혼잡윈도우가 18인 상황에서 Time-out이 발생하였으므로 임계값을 1/2로 줄이고(임계값 9), 혼잡윈도우는 1부터 다시 시작한다. (1로 감소하여 다시 시작한다.)

■ 혼잡 제어(Congestion Control)



- 처음에는 느린 출발(Slow Start) 기법을 적용한다.
- 혼잡윈도우를 1로 하여 전송을 시작한다.
- 정상적으로 ACK가 수신되면 혼잡윈도우를 1, 2, 4, 8, 16 형태로 증가시킨다.
- 임계값 16부터는 혼잡윈도우를 1, 2, 3씩 증가하면서 혼잡을 제어한다.
- 임계값 16이상 되면 혼잡 상황이 될 수 있기 때문에 임계값 이전에는 느린 출발(Slow Start) 기법을 이용하고, 그 이후에는 혼잡 회피 기법을 이용한다.

(1) TCP의 네트워크 혼잡(체증, Congestion) 인식과 용어

- Timeout 발생 : 심각한 혼잡으로 네트워크가 중단된 상태가 될 수 있는 상태로 인식한다.
- 3개의 중복 ACK 발생 : 경미한 혼잡 상태로 인식한다.
- 혼잡윈도우(Cwnd : Congestion window) : 한 번에 전송할 수 있는 세그먼트의 갯수이다.
세그먼트(TCP 헤더 + 데이터), 패킷(IP헤더 + TCP헤더 + 데이터)
- MMS(Maximum Segment Size) : 혼잡윈도우의 단위이다.
- 왕복시간(RTT: Round Trip Time) : 세그먼트가 목적지까지 도착하고 응답이 돌아오기까지 걸리는 시간이다. (1개의 세그먼트를 기준으로 시간을 정한다.)
- 임계값(ssthresh : slow start threshold) : 느린 출발(Slow Start) 기법을 사용하였을 경우 체증이 유발할 만한 지점이다. 수신측 버퍼에서 현재 수용할 수 있는 최대의 세그먼트의 갯수이다.

(2) TCP의 혼잡 제어 기법

•느린 출발(Slow Start)

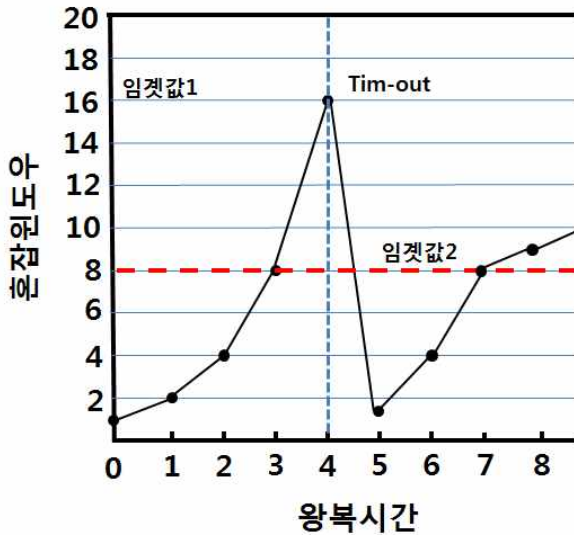
- 연결 시작 시에 혼잡윈도우(cwnd)를 최소값(예 : 1MMS)부터 전송을 시작한다.
- ACK가 수신되면 정상 수신되었다는 것이므로 혼잡윈도우(cwnd)를 증가시킨다.
- RTT마다 혼잡윈도우(cwnd)는 2배씩 증가한다. (1, 2, 4, 8, 16,... 지수적으로 증가한다.)

•혼잡 회피(Congestion Avoidance)

- 느린 출발(Slow Start) 구간에서 혼잡원도우는 지수적으로 증가하므로 어느 시점(임계값)에는 체증이 유발된다.
- 혼잡원도우의 임계값에 도달하거나 넘어갈 경우 세그먼트의 갯수를 선형적으로 조절하여 세그먼트의 갯수를 줄이는 기법이다.
- RTT마다 혼잡원도우(cnwd)가 1씩 증가한다. (1, 2, 3, 4, 5,... 선형적으로 증가한다.)

•빠른 복구(Fast Recovery)

- 경미한 혼잡 상황 시에 적용하는 방법이다.
- 정상 ACK이 수신되어 오류가 없다고 간주되면 느린 출발(Slow Start)을 생략하고, 혼잡 회피 단계로 진입시키는 기법이다.



- 처음에는 느린 출발(Slow Start) 기법을 적용한다.
- 혼잡원도우가 1MMS부터 전송을 시작하여 정상적으로 ACK가 수신되면 혼잡원도우를 1, 2, 4, 8, 16 형태로 증가시킨다.
- 그런데 임계값1이 16인 시점에서 Time-out 상태가 되었다.
- 임계값2를 1/2로 줄이고(6이므로 8로), 다시 혼잡원도우를 1MMS부터 다시 시작한다.
- 임계값2가 8이상 되면 혼잡 상황이 될 수 있기 때문에 임계값2이 8이전에는 느린 출발(Slow Start) 기법을 이용하고, 그 이후에는 혼잡 회피 기법을 이용한다.

[출제정보] 출제 정보 없음

[난이도] 상

문 17. 다중 프로그래밍 환경에서 연속 메모리 할당 방법에 대한 설명으로 옳지 않은 것은?

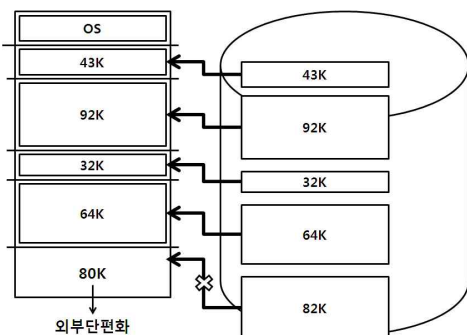
- ① 가변분할 메모리 할당은 프로세스의 크기에 따라 메모리를 나누는 것으로 단편화 문제가 발생하지 않는다.
- ② 가변분할 메모리 할당의 메모리 배치방법으로는 최초 적합, 최적 적합, 최악 적합 방법이 있다.
- ③ 고정분할 메모리 할당은 프로세스의 크기와 상관없이 메모리를 같은 크기로 나누는 것이다.
- ④ 고정분할 메모리 할당에서는 쓸모없는 공간으로 인해 메모리 낭비가 발생할 수 있다.

[정답] ①

[해설]

■ 가변 분할(동적 분할) 할당 기법

- 프로그램의 크기에 따라 주기억장치 분할 크기를 동적으로 분할하는 방식이다.



- 미리 크기를 결정하지 않고 실행할 프로세스의 크기에 맞게 기억 장소를 분할한다.

- 외부 단편화만 존재한다.
- 기억장소 활용율이 높아진다.
- 고정 분할 방식에 비해 실행될 프로세스 크기에 대한 제약이 완화된다.
- 동적 분할 방식은 고정 분할 방식에 비해 주기억장치의 크기를 넘지 않는 범위에서 큰 프로그램을 적재할 수 있다.

[출제정보] [출인원 1.03.03] 주기억장치-다중프로그래밍

[난이도] 하

문 18. 병렬 프로세서에 대한 설명으로 옳지 않은 것은?

- ① 프로세스 수준 병렬성은 다수의 프로세서를 이용하여 독립적인 프로그램 여러 개를 동시에 수행한다.
- ② 클러스터는 근거리 네트워크를 통하여 연결된 컴퓨터들이 하나의 대형 멀티 프로세서로 동작하는 시스템이다.
- ③ 공유 메모리 프로세서(SMP)는 단일 실제 주소 공간을 갖는 병렬 프로세서를 의미한다.
- ④ 각 프로세서의 메모리 접근법 분류에 따르면 UMA는 약결합형 다중처리 시스템, NUMA 및 NORMA는 강결합형 다중처리 시스템에 해당한다.

[정답] ④

[해설]

■ 병렬 프로세서(CPU)의 메모리(Memory) 접근 방법

(1) UMA(Uniform Memory Access, 균일 기억 장치 접근)

- 모든 프로세서들이 상호간에 연결되어 하나의 메모리를 공유하는 기술로 강결합형 다중처리 시스템에 해당한다.
- 각 프로세서가 메모리의 어느 곳이든 접근이 가능하다.
- 모든 프로세서가 메모리에 접근하는 시간은 동일하다
- 구조가 간단하다.

(2) NUMA(Non-Uniform Memory Access, 불균일 기억 장치 접근)

- UMA 모델을 보완하여 개선 시킨 기술로 약결합형 다중처리 시스템에 해당한다.
- 메모리 접근하는 시간이 프로세서와 메모리의 상대적인 위치에 따라 달라진다.
- 프로세서에 메모리나 입출력 장치들이 독립적으로 연결되어 있다.
- 각 프로세서가 메모리의 어느 곳이든 접근이 불가능하다.
- 모든 프로세서가 메모리에 접근하는 시간은 동일하지 않다.
- 구조가 복잡하다.

[출제정보] 출제 정보 없음

[난이도] 중

문 19. 다음 C 프로그램의 실행 결과로 옳은 것은?

```

01  #include <stdio.h>
02  int star = 10;
03  void printStar() {
04      printf("%d \n", star);
05  }
06  int main()
07  {
08      int star = 5;
09
10      printStar();
11      printf("%d \n", star);
12      return 0;
13  }
```

- ① 5
5
- ② 5
10
- ③ 10
5
- ④ 10
10

[정답] ③

[해설]

02 : 외부 변수 star를 정수형으로 선언하고, 초기값을 10을 주었다. 외부 변수 star는 모든 함수에 영향을 준다, 하지만 내부 변수로 선언(08 라인)한 동일한 변수가 있을 경우는 내부 변수가 우선권을 갖는다. 외부 변수는 번역 시에 확보되고, 프로그램이 종료되어야 소멸된다.

03 : printStar() 함수로 star의 값을 정수로 출력하는 함수이다.

06 : 프로그램의 시작점이다.

08 : 내부 변수 star가 선언되고 초기값 5를 주었다. int 앞에 auto가 생략된 형태이므로 스택(Stack)을 사용하며, 실행 시 생성된 후 함수 블록이 종료되면 소멸된다.

10 : printStar() 함수를 호출하였다. printStar() 함수는 star를 출력하므로 외부 변수에 영향을 받는다. 따라서 10이 출력된다. 만약 printStar() 함수 내에 int star = 5; 가 있었다면 5가 출력되지만 지금은 외부 변수의 영향을 받아 10을 출력된다.

11 : printStar() 함수를 수행하고 돌아와서 star를 출력했다. 이때는 내부 변수가 우선권이므로 내부 변수 star에 기억된 5을 출력한다.

[출제정보] [출인원 7.05.32] 기억 클래스(Storage Class)

[난이도] 중

문 20. 다음과 같이 P1, P2, P3, P4 프로세스가 동시에 준비 상태 큐에 도착했을 때 SJF(Shortest Job First) 스케줄링 알고리즘에서 평균 반환시간과 평균 대기시간을 바르게 연결한 것은? (단, 프로세스 간 문맥교환에 따른 오버헤드는 무시하며, 주어진 4개의 프로세스 외에 처리할 다른 프로세스는 없다고 가정한다)

프로세스	실행시간
P1	5
P2	6
P3	4
P4	9

평균 반환시간 평균 대기시간

- ① 6 6
- ② 6 7
- ③ 13 6
- ④ 13 7

[정답] ④

[해설]

평균 반환시간 = 평균 실행시간 + 평균 대기시간

평균 실행시간 = (5 + 6 + 4 + 9) / 4 = 24 / 4 = 6

평균 대기시간은

문제에서 동시에 모든 프로세스가 도착하였다고 하였고, SJF 스케줄링으로 처리하므로

실행시간이 짧은 프로세스부터 다음과 같이 배치시킨다.

P3	P1	P2	P4
4	5	6	7

P3의 대기시간 : 0

P1의 대기시간 : 4

P2의 대기시간 : (4+5) = 9

P4의 대기시간 : (4+5+6) = 15

평균 대기시간 = (0 + 4 + 9 + 15) / 4 = 28 / 4 = 7

따라서 평균 반환시간 = 6 + 7 = 13

[출제정보] [출인원 1.02.10] SJF(Short Job First) - 비선점형

[난이도] 중

연번	과목명	채형	1번	2번	3번	4번	5번	6번	7번	8번	9번	10번	11번	12번	13번	14번	15번	16번	17번	18번	19번	20번
29	컴퓨터일반	A	2	1	3	2	1	2	1	2	4	2	3	4	4	3	3	2	1	4	3	4