

[문제 1]

(1) Drill Down 또는 Drill-down

(2) Pivoting

※ 답안 작성 시 주의 사항 : 적어야 할 정답이 한 개인데, 정답이 여러 개 있는 경우가 있습니다. 그럴 경우 하나만 정확하게 적어야 합니다. 하나의 답만 요구하는 문제에 오답과 정답이 섞여 있을 경우 틀린 것으로 처리됩니다. 즉 (1)번은 Drill Down 또는 Drill-down 중 하나만 적으면 됩니다.

[문제 2]

10

5

4

3

2

※ 답안 작성 시 주의 사항 : 프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 결과를 한 줄로 10 5 4 3 2로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[디버깅]

i	j	a[i]	a[j]	temp	배열 a
0	1	3	4	3	
	2	4	3	4	
		4	10		
		10	4		
		10	2		
		10	5		
1	2	3	4	3	
	3	4	3	4	
		4	2		
		4	5		
		5	4		
2	3	3	2	3	
	4	3	4		
	5	4	3		
3	4	2	3	2	
	5	3	2		
4					

[문제 3]

Trigger 또는 트리거

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 없을 경우 한글 정답이나 영문 정답 모두 맞는 것으로 처리됩니다. 그리고 되도록이면 영문이나 한글 한 가지만 쓰세요. 두 가지를 쓸 경우 두 개 다 맞아야만 정답으로 인정됩니다. 한 가지 더 주의할 점은 영문 스펠링이나 한글 맞춤법이 틀렸을 경우 완전히 틀린 것으로 간주된다는 것을 잊지 마세요.

[문제 4]

다음 중 한 가지만 쓰면 됩니다.

SSO, Single Sign-On, Single Sign On, 싱글 사인온, 싱글 사인 온, OTP, One-Time Password, One Time Password, 일회용 패스워드

※ 답안 작성 시 주의 사항 : 한글 또는 영문을 Full-name이나 약어로 쓰라는 지시사항이 없을 경우 한글이나 영문 약어로 쓰는 것이 유리합니다. 영문을 Full-name으로 풀어쓰다가 스펠링을 쓸리면 오답으로 처리되니까요.

[문제 5]

다음 중 한 가지만 쓰면 됩니다.

DoS, Denial of Service, 서비스 거부 공격, DDoS, Distributed Denial of Service, 분산 서비스 거부 공격

[문제 6]

다음 중 한 가지만 쓰면 됩니다.

FCFS, First Come First Served, FIFO, First In First Out

[원문해석]

(FCFS) 스케줄링은 가장 간단한 스케줄링 알고리즘입니다. 이 방식에서는 CPU를 제일 처음 요청한 프로세스에게 CPU가 우선 할당됩니다.

(FCFS)는 기타 편파적인 특혜 없이 고객 또는 클라이언트의 요청이 도착하는 순서에 따라 수행하는 서비스 정책입니다. 이 정책은 판매 주문을 처리하거나 레스토랑의 좌석 배치를 결정할 때, 또는 택시 승차장 등에서 사용될 수 있습니다.

[문제 7]

다음 중 한 가지만 쓰면 됩니다.

SWOT 분석, SWOT analysis

[문제 8]

- 일관성 : 트랜잭션 수행이 성공적으로 완료되면 언제나 일관성 있는 데이터베이스 상태로 변환한다.
- 영속성 : 트랜잭션의 실행이 성공적으로 실행 완료된 후에는 시스템에 오류가 발생하더라도 트랜잭션에 의해 변경된 내용은 계속 보존되어야 한다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

[문제 9]

(1) 10000

(2) M/5

(3) TM[K]+PM[K]

※ 답안 작성 시 주의 사항 : <처리 조건>에 제시된 기준에 맞게 답안을 작성해야 합니다. 예를 들어 '사'에서 제시된 것처럼 배열이 요소를 작성할 때 TM[K]과 같이 대괄호를 이용해서 작성해야 합니다. TM(K)과 같이 중괄호를 이용해 작성하지 않도록 유의하세요.

[디버깅]

※ 다음과 같이 자료가 입력되었다고 가정하고 디버깅한 결과입니다.

<자 료>

임철식 537620

양동수 538973

이동훈 173105

QUIT

N	P	T	M	SW	K	배열	출력	
임철식	537620	537620	10000	1	1	PM	<div>출장비 지급 일람표</div> <div>성명출장비만원오천원천원오백원백원오십원십원오원일원</div> <div>임철식53762053121102000</div>	
		7620	5000	0	2	<div>53121102000</div>		
		2620	1000	1	3	TM		
		620	500	0	4			
		120	100	1	5			
		20	50	0	6			
		20	10	1	7			<div>53121102000</div>
		0	5	0	8			
		0	1	1	9			
		양동수	538973	538973	10000			1
8973	5000			0	2	<div>53131412003</div>		
3973	1000			1	3	TM		
973	500			0	4			
473	100			1	5			
73	50			0	6			
23	10			1	7		<div>106252514003</div>	
3	5			0	8			
3	1			1	9			
이동훈	173105			173105	10000		1	1
		3105	5000	0	2	<div>17030100100</div>		
		3105	1000	1	3	TM		
		105	500	0	4			
		105	100	1	5			
		5	50	0	6			
		5	10	1	7		<div>123282614113</div>	
		5	5	0	8			
		0	1	1	9			
		QUIT						

[문제 10]

다음 중 한 가지만 쓰면 됩니다.

Voice over Internet Protocol, VoIP, VOIP

※ 답안 작성 시 주의 사항 : 영문을 Full-name 또는 약어로 쓰라는 지시사항이 있을 경우 Full-name 보다 는 약어를 쓰는 것이 유리합니다. Full-name으로 풀어쓰다가 스펠링을 틀리면 오답으로 처리되니까요.

[원문해석]

이것은 전통적인 PSTN 회선 전송망이 아니라 IP를 사용해 음성 데이터를 패킷 형태로 송신하는 것을 통해 사용자가 인터넷을 전화 통화용 송신매체로 사용할 수 있도록 하는 하드웨어 및 소프트웨어 분류를 말한다.

[문제 11]

(1) b

(2) d

(3) a

(4) c

(5) e

※ 답안 작성 시 주의 사항 : 답안 작성 요령이 제시된 경우에는 답안 작성 요령에 맞게 작성해야 합니다. 예를 들어 (1) 답을 보기에서 찾아 a, b, c, d, e 중 하나로 답하라고 했는데, “자기고과실시”라고 답을 작성한 경우 오답으로 처리될 수 있으니 주의하세요.

[문제 12]

INSERT INTO 학생 VALUES(98170823, '한국산', 3, '경영학개론', '?-1234-1234');

※ 답안 작성 시 주의 사항 : SQL문은 대·소문자 구분없이 작성해도 관계없습니다. 단, 한 글자라도 틀리면 오답으로 처리됩니다. 특히 문제에 제시된 답안 작성 요령대로 문자열 구분은 작은따옴표(' ')로 하고 문장 끝에는 반드시 세미콜론(;)을 붙여야 한다는 것을 주의하세요.

[문제 13]

다음 중 한 가지만 쓰면 됩니다.

Pharming, 파밍

[문제 14]

다음 중 한 가지만 쓰면 됩니다.

Nomophobia, 노모포비아

[문제 15]

As-Is 분석이 현재의 업무 프로세스를 분석하는 것을 의미한다면 To-Be 분석은 미래에 구현하고자 하는 업무 프로세스를 정의하는 분석이다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

[문제 16]

num[i]

※ 답안 작성 시 주의 사항 : C언어 코드 형식으로 쓰라는 조건이 있으므로 C언어 코드 형식에 맞게 답안을 작성해야 합니다. 예를 들어, C언어의 변수명은 대소문자를 구분하므로 num[i]를 NUM[I]와 같이 대문자로 작성하면 오답으로 처리됩니다.

[디버깅]

※ num 배열에 다음과 같이 순서대로 입력되었다고 가정하고 디버깅한 결과입니다.

num[10]	70	82	43	90	65	55	95	45	68	72
---------	----	----	----	----	----	----	----	----	----	----

i	num[i]	min	출력
0	70	9999	가장 작은 값은 43이다.
1	82	70	
2	43	43	
3	90		
4	65		
5	55		
6	95		
7	45		
8	68		
9	72		
10			

[문제 17]

알고리즘의 소요 시간을 정량화하기 위해 알고리즘이 처리해야 할 연산의 수가 증가할 때 시간이 증가하는 대략적인 패턴을 수치화한 것이다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

[문제 18]

다음 중 한 가지만 쓰면 됩니다.

GLONASS, 글로나스

[문제 19]

CASCADE

※ 답안 작성 시 주의 사항 : SQL문은 소문자로 작성해도 관계없습니다. 단 SQL 문은 한 글자라도 틀리면 오답으로 처리됩니다. 특히 문장 끝에 있는 세미콜론(;)을 조심하세요. 이번 문제는 괄호 밖에 세미콜론이 있으므로 정답에 세미콜론을 붙이면 안 됩니다.

[문제 20]

다음 중 한 가지만 쓰면 됩니다.

Augmented Reality, AR, Mixed Reality, MR

기출문제 정답

[문제 1]

number % i == 0

※ 답안 작성 시 주의 사항 : C언어 코드 형식으로 쓰라는 조건이 있으므로 C언어 코드 형식에 맞게 답안을 작성해야 합니다. 예를 들어, C언어의 변수명은 대소문자를 구분하므로 number를 NUMBER와 같이 대문자로 작성하거나 “같다”는 의미의 관계 연산자를 ‘==’이 아닌 ‘=’으로 작성하면 오답으로 처리됩니다.

[디버깅]

number (main)	cnt	i (main)	number (isprime)	i (isprime)	반환값 (isprime)	출력
100	0	2	2	2	1	100을 넘지 않는 소수는 25개입니다.
	1					
	2	3	3	2		
				3	1	
		4	4	2	0	
	3	5	5	2		
				3		
				4		
				5	1	
		6	6	2	0	
	4	7	7	2		
				3		
				4		
				5		
				6		
				7	1	
		8	8	2	0	
		9	9	2		
				3	0	
		10	10	2	0	
	⋮	⋮	⋮	⋮	⋮	
		96	96	2	0	
	25	97	97	2		
				3		
				⋮		
				⋮		
				97	1	
		98	98	2	0	
		99	99	2		
				3	0	
		100				

[문제 2]

다음 중 한 가지만 쓰면 됩니다.

캡슐화, Encapsulation

[문제 3]

30

[디버깅]

a	sum	a%2	출력
0	0		30
1		1	
2	2	0	
3		1	
4	6	0	
5		1	
6	12	0	
7		1	
8	20	0	
9		1	
10	30	0	

[문제 4]

다음 중 한 가지만 쓰면 됩니다.

Stuxnet, 스텍스넷

[문제 5]

• 원인 : 테이블에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하고, 이 중복(Redundancy)으로 인해 테이블 조작 시 문제가 발생하는 현상

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

• 종류

- ▶ 삽입 이상(Insertion Anomaly)
- ▶ 삭제 이상(Deletion Anomaly)
- ▶ 갱신 이상(Update Anomaly)

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 없을 경우 한글 정답이나 영문 정답 모두 맞는 것으로 처리됩니다. 그리고 되도록이면 영문이나 한글 한 가지만 쓰세요. 두 가지를 쓸 경우 두 개 다 맞아 야만 정답으로 인정됩니다. 한 가지 더 주의할 점은 영문 스펠링이나 한글 맞춤법이 틀렸을 경우 완전히 틀린 것으로 간주된다는 것을 잊지 마세요.

[문제 6]

다음 중 한 가지만 쓰면 됩니다.

ad-hoc, ad-hoc 네트워크, 애드혹

[문제 7]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- (1) 메타 데이터, Metadata
- (2) MDR, Meta Data Registry, 메타 데이터 레지스트리

[문제 8]

- (1) N
- (2) S

[디버깅]

※ 아래의 디버깅 표는 4부터 10까지의 정수만을 대상으로 디버깅한 결과입니다.

TC	N	TOT	L	S	R	출력
0	4	0 1 3	2	1 2 3	0 0	1
	5	0 1	2	1 2 3	0 1	
1	6	0 1 3 6	3	1 2 3 4	0 0 0	
	7	0 1	3	1 2 3 4	0 1 1	
	8	0 1 3 7	4	1 2 3 4 5	0 0 2 0	
	9	0 1 4	4	1 2 3 4 5	0 1 0 1	
	10	0 1 3 8	5	1 2 3 4 5 6	0 0 1 2 0	
	11					

[문제 9]

다음 중 한 가지만 쓰면 됩니다.

증분, Incremental

[문제 10]

COMMIT 연산은 트랜잭션 처리가 정상적으로 종료되어 트랜잭션이 수행한 변경 내용을 데이터베이스에 반영하는 연산이다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

[문제 11]

요구 사항 분석, 개념적 설계, 논리적 설계, 물리적 설계, 구현

[문제 12]

다음 중 한 가지만 쓰면 됩니다.

목표 복구 시간, RTO, Recovery Time Objective

[문제 13]

- (1) SCP
- (2) SCE

※ 답안 작성 시 주의 사항 : 영문 약어로 쓰라는 지시사항이 있으므로 영문 약어로 작성해야 합니다.

[문제 14]

다음 중 한 가지만 쓰면 됩니다.

AI, Artificial Intelligence

※ 답안 작성 시 주의 사항 : 영문을 Full-name 또는 약어로 쓰라는 지시사항이 있을 경우 Full-name 보다 는 약어를 쓰는 것이 유리합니다. Full-name으로 풀어쓰다가 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

(인공 지능)은 특히 컴퓨터 시스템에 의한 인간 지능 프로세스의 시뮬레이션이다. 이러한 과정에는 학습, 추론 및 자체 연결이 포함된다. (인공 지능)의 특별한 응용 프로그램은 전문가 시스템, 음성 인식, 머신 비전 등을 포함한다. (인공 지능)은 컴퓨터가 인간처럼 행동하는 것과 관련된 컴퓨터 과학의 영역이다. 이 용어는 1956년에 메사추세츠 공과 대학의 존 매커시에 의해 만들어졌다. 인공 지능은 게임, 전문가 시스템, 자연 언어, 신경 회로, 로봇 공학을 포함한다.

[문제 15]

다음 중 한 가지만 쓰면 됩니다.

OLAP, Online Analytical Processing

[문제 16]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- (1) Field
- (2) DBMS, Database Management System
- (3) DBA, Database Administrator

[원문해석]

전통적인 데이터베이스는 (필드), 레코드, 파일로 구성된다. (필드)는 단일 정보 조각을 의미하며, 레코드는 필드들로 구성된 한 개의 완성된 집합이다. 파일은 레코드들의 모음이다. (DBMS)는 데이터베이스로부터 정보를 저장하고 수정하며 추출하도록 해주는 프로그램들의 집합이다. (DBA)는 (DMBS)에서 최고 수준의 권한을 갖는다. (DBA)는 자신이 가진 권한의 일정 부분을 다른 사람에게 허용할 수 있으며, 또한 어느 때라도 이를 철회할 수 있는 능력을 보유하고 있다.

[문제 17]

- 기밀성은 시스템 내의 정보와 자원은 인가된 사용자에게만 접근이 허용되는 것을 의미한다.
- 무결성은 시스템 내의 정보는 인가된 사용자만 수정이 가능한 것을 의미한다.
- 가용성은 인가받은 사용자는 언제라도 사용 가능한 것을 의미한다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 설명이라는 것이 판단될 정도면 됩니다.

[문제 18]

DELETE FROM 학생 WHERE 이름 = 'Scott';

※ 답안 작성 시 주의 사항 : SQL문은 대·소문자 구분없이 작성해도 관계없습니다. 단, 한 글자라도 틀리면 오답으로 처리됩니다. 특히 문제에 제시된 답안 작성 요령대로 문자열 구분은 작은따옴표(' ')로 하고 문장 끝에는 반드시 세미콜론(;)을 붙여야 한다는 것을 주의하세요.

[문제 19]

- (1) 의사 결정 지원 시스템, DSS, Decision Support System
- (2) 전략 정보 시스템, SIS, Strategic Information System

※ 문제의 지문이 긴 관계로 일부만 복원이 되었습니다. 실제 시험에서는 총 3문항이 출제되었으며 세 번째 문항은 지문을 읽고 제시된 보기에서 하나를 골라 작성하는 문제였는데, 세 번째 문항의 답은 제시된 지문에서 쉽게 답을 찾을 수 있었습니다.

기출문제 정답

[문제 1]

234

모든 C 프로그램은 반드시 main() 함수부터 시작해야 합니다.

```
int main( ) {  
  ❶ int result;  
  ❷ result = res200( );  
  printf("%d\n", result);  
}
```

❶ 정수형 변수 result를 선언합니다.

❷ 인수 없이 res200 함수를 호출한 다음 돌려받은 값을 result에 저장한다. res200() 함수를 호출합니다.

```
int res200( ) {  
  ❶ return 200 + res30( );  
}
```

❶ 200에 res30 함수를 호출한 다음 돌려받은 값을 더해서 함수를 호출한 곳(main 함수)으로 반환합니다. res30() 함수를 호출합니다.

```
int res30( ) {  
  ❶ return 30 + res10( );  
}
```

❶ 30에 res10 함수를 호출한 다음 돌려받은 값을 더해서 함수를 호출한 곳(res200 함수)으로 반환합니다. res10() 함수를 호출합니다.

```
int res10( ) {  
  ❶ return 4;  
}
```

❶ 반환값 4를 가지고 res10() 함수를 호출했던 res30() 함수로 제어를 옮깁니다.

```
int res30( ) {  
  ❶ return 30 + res10( );  
}
```

❶ 반환값 34(30+4)를 가지고 res30() 함수를 호출했던 res200() 함수로 제어를 옮깁니다.

```
int res200( ) {  
  ❶ return 200 + res30( );  
}
```

❶ 반환값 234(200+34)를 가지고 res200() 함수를 호출했던 main() 함수로 제어를 옮깁니다.

```
int main( ) {  
  int result;  
  ❶ result = res200( );  
  ❷ printf("%d\n", result);  
}
```

❶ res200() 함수에서 돌려받은 234를 result에 저장합니다.

② result의 값 234를 정수형으로 출력한 후 커서를 다음 줄 처음으로 옮깁니다.

결과 234

[문제 2]

특정 분야에서 우수한 상대를 모델로 하여 자기 기업과의 성과 차이를 비교 분석하고, 단점을 극복하기 위해 상대의 앞선 운영프로세스를 배우면서 꾸준히 노력하여 자기혁신을 하는 경영기법이다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 3]

- 데이터 마이닝의 개념 : 데이터 웨어하우스에 저장된 데이터 집합에서 사용자의 요구에 따라 유용하고 가 능성 있는 정보를 발견하기 위한 기법이다.

- 군집화의 개념 : 상호 간에 유사한 특성을 갖는 데이터들을 집단화(Clustering)하는 방법이다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 4]

a[i]

※ 답안 작성 시 주의 사항 : 배열의 요소를 지정할 때는 반드시 a[i]와 같이 대괄호를 사용해야 합니다. a(i)와 같이 소괄호를 사용하지 않도록 주의하세요.

[디버깅]

i	max	min	출력
0	10	10	90
1	30		10
2	50		
3	70		
4	90		
5			

[문제 5]

다음 중 한 가지만 쓰면 됩니다.

래드섹, RadSec, 래드섹 프로토콜, RadSec Protocol

※ 답안 작성 시 주의 사항 : 한글 또는 영문으로 쓰라는 지시사항이 없을 경우 한글 정답이나 영문 정답 모두 맞는 것으로 처리됩니다. 그리고 한글이나 영문 두 가지를 쓸 경우 두 개 다 맞아야만 정답으로 인정되므로 영문이나 한글 한 가지만 쓰는 것이 좋습니다. 영문 스펠링이나 한글 맞춤법이 틀렸을 경우 완전히 틀린 것으로 간주된다는 것도 잊지 마세요.

[문제 6]

다음 중 한 가지만 쓰면 됩니다.

블록체인, blockchain

[문제 7]

- **병행 제어의 개념** : 다중 프로그램의 이점을 활용하여 동시에 여러 개의 트랜잭션을 병행 수행할 때 실행되는 트랜잭션들이 데이터베이스의 일관성을 파괴하지 않도록 트랜잭션 간의 상호 작용을 제어하는 기술이다.
 - **로킹 기법의 개념** : 주요 데이터의 액세스를 상호 배타적으로 하는 것으로, 트랜잭션들이 어떤 로킹 단위를 액세스하기 전에 Lock(잠금)을 요청해서 Lock이 허락되어야만 그 로킹 단위를 액세스할 수 있도록 하는 기법이다.
- ※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 8]

- (①) H
- (②) Q[L],[Y]
- (③) P[L],[Y]

[디버깅]

X	Y	H	L	배열																									
1	1 2 3 4 5	0 2 4 6 8 10	0	<p>P 배열</p> <table><tr><td>2</td><td>4</td><td>6</td><td>8</td><td>10</td></tr><tr><td>12</td><td>14</td><td>16</td><td>18</td><td>20</td></tr><tr><td>22</td><td>24</td><td>26</td><td>28</td><td>30</td></tr><tr><td>32</td><td>34</td><td>36</td><td>38</td><td>40</td></tr><tr><td>42</td><td>44</td><td>46</td><td>48</td><td>50</td></tr></table>	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50
2	4	6	8		10																								
12	14	16	18		20																								
22	24	26	28		30																								
32	34	36	38		40																								
42	44	46	48	50																									
2	1 2 3 4 5	12 14 16 18 20																											
3	1 2 3 4 5	22 24 26 28 30																											
4	1 2 3 4 5	32 34 36 38 40																											
5	1 2 3 4 5	42 44 46 48 50																											
2	1	1 2 3 4 5	5 4 3 2 1	<p>Q 배열</p> <table><tr><td>42</td><td>32</td><td>22</td><td>12</td><td>2</td></tr><tr><td>44</td><td>34</td><td>24</td><td>15</td><td>4</td></tr><tr><td>46</td><td>36</td><td>26</td><td>16</td><td>6</td></tr><tr><td>48</td><td>38</td><td>28</td><td>18</td><td>8</td></tr><tr><td>50</td><td>40</td><td>30</td><td>20</td><td>10</td></tr></table>	42	32	22	12	2	44	34	24	15	4	46	36	26	16	6	48	38	28	18	8	50	40	30	20	10
42	32	22	12		2																								
44	34	24	15		4																								
46	36	26	16		6																								
48	38	28	18		8																								
50	40	30	20	10																									
	2	1 2 3 4 5	5 4 3 2 1																										
	3	1 2 3 4 5	5 4 3 2 1																										
	4	1 2 3 4 5	5 4 3 2 1																										
	5	1 2 3 4 5	5 4 3 2 1																										

1	1	1 2 3 4 5	5 4 3 2 1	<div>P 배열</div> <table><tr><td>50</td><td>48</td><td>46</td><td>44</td><td>42</td></tr><tr><td>40</td><td>38</td><td>36</td><td>34</td><td>32</td></tr><tr><td>30</td><td>28</td><td>26</td><td>24</td><td>22</td></tr><tr><td>20</td><td>18</td><td>16</td><td>14</td><td>12</td></tr><tr><td>10</td><td>8</td><td>6</td><td>4</td><td>2</td></tr></table>	50	48	46	44	42	40	38	36	34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2
50	48	46	44		42																								
40	38	36	34		32																								
30	28	26	24		22																								
20	18	16	14		12																								
10	8	6	4	2																									
	2	1 2 3 4 5	5 4 3 2 1																										
	3	1 2 3 4 5	5 4 3 2 1																										
	4	1 2 3 4 5	5 4 3 2 1																										
	5	1 2 3 4 5	5 4 3 2 1																										

[문제 9]

÷

[문제 10]

1024

```
int main( ) {
  ❶ printf("%d\n", power(2, 10));
  return 0;
}
```

- ❶ 2와 10을 인수로 하여 power 함수를 호출한 다음 돌려받은 값을 정수형으로 출력합니다. power(2, 10)으로 호출합니다.

```
int power(int data, int exp) {
  ❶ int i, result = 1;
  ❷ for(i = 0; i < exp; i++)
  ❸   result = result * data;
  ❹ return result;
}
```

power() 함수가 호출될 때 2와 10이 전달되었으므로 data는 2이고 exp는 10입니다.

- ❶ 정수형 변수 i와 result를 선언하고, result의 초기값으로 1을 할당합니다.
 ❷ 반복 변수 i가 0에서 시작하여 1씩 증가하면서 exp보다 작은 동안 ❸번 문장을 반복합니다. exp가 10을 가지고 있으므로 총 10회 반복 수행합니다.
 ❸ result * data의 결과를 result에 저장합니다.
 ❷~❸번을 10회 반복하는 과정은 다음과 같습니다.

i	result	data	exp
0	1	2	10
	2		
1	4		
2	8		
3	16		
4	32		
5	64		
6	128		
7	256		
8	512		
9	1024		
10			

- ④ result의 값이 1024이므로 반환값 1024를 가지고 power(2, 10) 함수를 호출했던 main() 함수로 제어를 옮깁니다.

```
int main( ) {
    ❶ printf("%d\n", power(2, 10));
    ❷ return 0;
}
```

- ❶ power(2, 10) 함수에서 돌려받은 1024를 정수형으로 출력한 후 커서를 다음 줄 처음으로 옮깁니다.
결과 **1024**
- ❷ main() 함수에서 리턴 값으로 0을 반환하는 것은 에러 없이 정상적으로 프로그램이 종료되었다는 의미입니다.

[문제 11]

- (①) Not Null
- (②) Primary
- (③) Foreign
- (④) References

CREATE TABLE 학생 (이름 VARCHAR(15) Not Null, 학번 CHAR(8), 전공 CHAR(5), 성별 SEX, 생년월일 DATE, Primary KEY(학번), Foreign KEY(전공) References 학과(학과코드) ON DELETE SET NULL ON UPDATE CASCADE, CONSTRAINT 생년월일제약 CHECK(생년월일 >='1980-01-01'));	<학생> 테이블을 생성한다. 이름 속성은 최대 문자 15자로, NULL 값을 갖지 않는다. 학번 속성은 문자 8자이다. 전공 속성은 문자 5자이다. 성별 속성은 'SEX' 도메인을 자료형으로 사용한다. 생년월일 속성은 DATE 자료형을 갖는다. 학번 속성을 기본키로 정의한다. 전공 속성은 <학과> 테이블의 학과코드 속성을 참조하는 외래키이다. <학과> 테이블에서 튜플이 삭제되면 관련된 모든 튜플의 전공 속성의 값을 NULL로 변경한다. <학과> 테이블에서 학과코드가 변경되면 관련된 모든 튜플의 전공 속성의 값도 같은 값으로 변경한다. 생년월일 속성에는 1980-01-01 이후의 값만을 저장할 수 있으며, 이 제약 조건의 이름은 '생년월일제약'이다.
---	--

[문제 12]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- (①) Semantic Web, 시맨틱 웹

- (②) TPS, Transaction Processing System, 거래 처리 시스템
- (③) CALS, Commerce At Light Speed, 광속상거래

※ 문제의 지문이 긴 관계로 일부만 복원이 되었습니다. 실제 시험에서는 총 4문항이 출제되었으며 네 번째 문항은 지문을 읽고 제시된 보기에서 하나를 골라 작성하는 문제였는데, 네 번째 문항의 답은 제시된 지문에서 쉽게 답을 찾을 수 있었습니다.

[문제 13]

- **비정규화의 개념** : 정규화로 인해 여러 개로 분해된 릴레이션들에서 원하는 정보를 얻기 위해서는 조인을 사용하여 다시 연결해야 하는데, 지나치게 자주 조인을 사용하면 응답 속도가 떨어지므로 정규화에 위배되지만 성능 향상을 위해 다시 테이블을 합쳐야 한다. 이것을 비정규화라고 한다.

※ **답안 작성 시 주의 사항** : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 14]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- (①) 사회 공학
- (②) 살라미
- (③) 다크 데이터

※ 문제의 지문이 긴 관계로 일부만 복원이 되었습니다.

※ **답안 작성 시 주의 사항** : 영문 약어나 한글로 쓰라는 지시사항이 있으므로 반드시 영문 약어나 한글로 작성해야 합니다.

[문제 15]

다음 중 한 가지만 쓰면 됩니다.

SDN, Software Defined Networking, 소프트웨어 정의 네트워킹

※ **답안 작성 시 주의 사항** : 한글 또는 영문으로 쓰라는 지시사항이 없을 경우 한글 정답이나 영문 정답 모두 맞는 것으로 처리됩니다. 그리고 한글이나 영문 두 가지를 쓸 경우 두 개 다 맞아야만 정답으로 인정되므로 영문이나 한글 한 가지만 쓰는 것이 좋습니다. 영문 스펠링이나 한글 맞춤법이 틀렸을 경우 완전히 틀린 것으로 간주된다는 것도 잊지 마세요.

[문제 16]

다음 중 한 가지만 쓰면 됩니다.

웹 서비스 기술 언어, WSDL, Web Services Description Language

[문제 17]

- (①) : multimedia
- (②) : text
- (③) : image

※ **답안 작성 시 주의 사항** : 영문을 Full-name으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

MMS는 셀룰러 네트워크를 통해 핸드폰으로 또는 핸드폰으로부터 (멀티미디어) 콘텐츠가 포함된 메시지를

보내는 표준적인 방법이다. MMS 표준은 SMS(Short Message Service) 기능을 확장하여, 160자 이상의 (문자) 메시지를 교환 할 수 있도록 한다. (텍스트) 전용 SMS와 달리, MMS는 최대 40초 분량의 비디오, 하나의 (이미지), 여러 (이미지)들의 슬라이드 쇼 또는 오디오를 포함한 다양한 미디어를 전달 할 수 있다.

기출문제 정답

[문제 1]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

① 결정자, Determinant

② 종속자, Dependent

③ 어떤 애트리뷰트(속성)가 기본키에 대해 완전히 종속적인 경우 완전 함수 종속이라고 한다.

※ ③ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 2]

Select 학번, 이름 From 학생 Where 학년 in (3, 4);

※ 답안 작성 시 주의 사항 : 대·소문자를 구분하지 않습니다. 단 스펠링이 하나라도 틀렸을 경우 부분 점수 1도 없는 오답으로 처리된다는 것을 잊지 마세요.

[문제 3]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

① DDL 컴파일러, DDL 처리기, DDL 번역기

② 트랜잭션 관리자

[문제 4]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

① 시스템, System

② 덤프, Dump

③ 검사점, Check Point

[문제 5]

① MAX_STACK_SIZE-1 또는 9

② item

③ -1

④ top--

※ 답안 작성 시 주의 사항 : C언어에서 사용하는 변수명은 대소문자를 구분하기 때문에 변수명을 작성할 때는 대소문자를 구분해서 정확히 작성해야 합니다.

[디버깅]

top (전역변수)	item (push 함수)	e (main 함수)	stack[top] (전역변수)	return (isempty 함수)	출력
-1					
0	20		20		
1	30		30		
2	40		40		
					stack's status
2		40		0	value = 40
1		30		0	value = 30
0		20		0	value = 20
-1				1	

[문제 6]

① 95, 75, 85, 100, 50

② E[j]

※ 답안 작성 시 주의 사항 : 배열의 요소를 지정할 때는 반드시 E[j]와 같이 대괄호를 사용해야 합니다. E(j)와 같이 소괄호를 사용하지 않도록 주의하세요.

[디버깅]

아래 디버깅 표는 정렬이 수행되는 과정만 디버깅한 결과입니다.

i	j	Temp	E[i]	E[j]	E 배열					
0	0	0	95	95						
	1	95	75	75						
	2	75	50	95	<table><tr><td>95</td><td>75</td><td>85</td><td>100</td><td>50</td></tr></table>	95	75	85	100	50
	95	75	85	100	50					
	3			85	<table><tr><td>75</td><td>95</td><td></td><td></td><td>75</td></tr></table>	75	95			75
	75	95			75					
4			100	<table><tr><td>50</td><td></td><td></td><td></td><td></td></tr></table>	50					
50										
5			50	<table><tr><td></td><td></td><td></td><td></td><td></td></tr></table>						
			75							
1	1	95	95	85						
	2	85	85	95	<table><tr><td>50</td><td>95</td><td>85</td><td>100</td><td>75</td></tr></table>	50	95	85	100	75
	50	95	85	100	75					
	3		75	100	<table><tr><td></td><td>85</td><td>95</td><td></td><td>85</td></tr></table>		85	95		85
		85	95		85					
	4			75	<table><tr><td></td><td>75</td><td></td><td></td><td></td></tr></table>		75			
	75									
5			85							
2	2	95	95	100						
	3		85	85	<table><tr><td>50</td><td>75</td><td>95</td><td>100</td><td>85</td></tr></table>	50	75	95	100	85
	50	75	95	100	85					
	4			95	<table><tr><td></td><td></td><td>85</td><td></td><td>95</td></tr></table>			85		95
		85		95						
5										
3	3		100	95	<table><tr><td>50</td><td>75</td><td>85</td><td>100</td><td>95</td></tr></table>	50	75	85	100	95
	50	75	85	100	95					
	4				<table><tr><td></td><td></td><td></td><td>95</td><td>100</td></tr></table>				95	100
				95	100					
5										
4										

[문제 7]

① J+S

② N

③ K-1

④ N+1

⑤ i+S

[디버깅]

S	K	i	J	P	N	배열 A																									
1	5	1	0	1	0	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr><tr><td>16</td><td>17</td><td>18</td><td>19</td><td>6</td></tr><tr><td>15</td><td>24</td><td>25</td><td>20</td><td>7</td></tr><tr><td>14</td><td>23</td><td>22</td><td>21</td><td>8</td></tr><tr><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td></tr></table>	1	2	3	4	5	16	17	18	19	6	15	24	25	20	7	14	23	22	21	8	13	12	11	10	9
			1	2	3		4	5																							
			16	17	18		19	6																							
			15	24	25		20	7																							
			14	23	22		21	8																							
			13	12	11		10	9																							
	1	2	1																												
	2	3	2																												
	3	4	3																												
4	5	4																													
5	6	5																													
4	2			1	6																										
				2	7																										
				3	8																										
				4	9																										
				5																											
-1			4	1	10																										
			3	2	11																										
			2	3	12																										
			1	4	13																										
				5																											
			3	4		1	14																								
	2	15																													
	3	16																													
				4																											
1			2	1	17																										
			3	2	18																										
			4	3	19																										
				4																											
	2	3		1	20																										
				2	21																										
	4		3																												
-1			3	1	22																										
			2	2	23																										
				3																											
	1	3		1	24																										
				2																											
1	0		3	1	25																										
				2																											

[문제 8]

① 대차대조표

- ② 이익잉여금 처분 계산서
- ③ 현금 흐름표
- ④ 안전성
- ⑤ 매출채권회전을

[문제 9]

네트워크의 중간에서 남의 패킷 정보를 도청하는 해킹 유형의 하나로 수동적 공격에 해당한다.

※ 답안 작성 시 주의 사항 : 답안 내용이 위와 완전히 일치할 수는 없습니다. 작성된 답안이 요구하는 것에 대한 서술이라는 것이 판단될 정도면 됩니다.

[문제 10]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① OWASP, The Open Web Application Security Project
- ② 그레이웨어, grayware
- ③ 공통평가기준, CC, Common Criteria
- ④ 아이핀, i-PIN, internet Personal Identification Number, 인터넷 개인식별번호
- ⑤ 타이포스쿼팅, typosquatting

[문제 11]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① QoS, Quality of Service, 서비스 품질
- ② AJAX, Asynchronous JavaScript and XML, 비동기 자바 스크립트와 XML
- ③ 크리슈머, Cresumer
- ④ 저전력 블루투스 기술, BLE, Bluetooth Low Energy
- ⑤ Infolust, 정보 열광자, 인포러스트

[문제 12]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① OOP, Object-oriented programming
- ② object
- ③ method
- ④ message
- ⑤ JAVA

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

① 소프트웨어가 기계의 기계적인 부품들인 것처럼 객체가 조립되고 생성될 수 있도록 현실 세계의 개체(Entity)를 기계 부품과 같은 하나의 객체로 만드는 기법이다. 구조적인 문제들로 인한 소프트웨어 위기의 해결책으로 사용된다. 소프트웨어를 재사용하고 확장하여 고품질 소프트웨어를 신속하게 개발하고 유지하기 용이하다

② 데이터와 데이터를 처리하는 함수를 캡슐화한 하나의 소프트웨어 모듈이다. 이것은 상태와 행위를 가지고 있으며, 다른 (객체)들과 구별될 수 있는 이름과 일정한 기억장소를 갖는다.

③ 객체에 수행될 연산, 즉 객체의 속성값에 대한 변경이나 검색을 기술한 것이다. 또한 특정한 이것을 수행하기 위해서는 해당 객체에 메시지를 보내야 한다.

④ 객체들 간에 상호작용을 하는 데 사용되는 수단으로, 객체에게 어떤 행위를 하도록 지시하는 명령 또는 요구 사항이다. (메시지)를 받는 수신자의 이름, 객체가 수행할 메소드 이름, 메소드를 수행할 때 필요한 속성값으로 구성된다. 이것을 받은 객체는 요구된 메소드를 수행하여 결과를 반환하게 된다.

⑤ 보안성이 뛰어나고 유지 보수가 용이한 객체 지향 프로그래밍(OOP) 언어이다. 컴파일 한 코드를 다른 운영체제에서 사용할 수 있도록 클래스를 제공함으로써 뛰어난 이식성을 갖고 있다. 또한 보다 효율적인 프로그래밍이 가능하도록 분산 환경을 지원한다.

기출문제 정답

[문제 1]

- ① C
- ② K
- ③ R
- ④ TR
- ⑤ -1 또는 (-1)

[디버깅 표]

TR	START	END	SW	R	C	K	SON 배열																									
0	1	5	1	1	1	0	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr><tr><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td></tr><tr><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr><tr><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr><tr><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td></tr></table>	1	2	3	4	5	10	9	8	7	6	11	12	13	14	15	20	19	18	17	16	21	22	23	24	25
					1	2		3	4	5																						
					10	9		8	7	6																						
					11	12		13	14	15																						
					20	19		18	17	16																						
					21	22		23	24	25																						
2	1																															
3	2																															
4	3																															
5	4																															
6	5																															
1	5	1	-1	2	5	6																										
					4	7																										
					3	8																										
					2	9																										
					1	10																										
					0																											
5	1	5	1	3	1	11																										
					2	12																										
					3	13																										
					4	14																										
					5	15																										
					6																											
1	5	1	-1	4	5	16																										
					4	17																										
					3	18																										
					2	19																										
					1	20																										
					0																											
5	1	5	1	5	1	21																										
					2	22																										
					3	23																										
					4	24																										
					5	25																										
					6																											
1	5	1	-1	6																												

[문제 2]

- ① commit
- ② rollback
- ③ grant
- ④ revoke
- ⑤ cascade

※ 답안 작성 시 주의 사항 : 대·소문자를 구분하지 않습니다. 단 스펠링이 하나라도 틀렸을 경우 부분 점수 1도 없는 오답으로 처리된다는 것을 잊지 마세요.

[문제 3]

- ① AS-IS
- ② 사무국
- ③ 서면평가

[문제 4]

!= 또는 >

※ 답안 작성 시 주의 사항 : C언어에서 사용하는 형식에 맞게 정확히 작성해야 합니다. '같지 않다'는 표현을 '<>'로 하지 않도록 주의하세요.

[디버깅]

배열 a에 차례대로 1, 2, 3, 4, 5가 입력되었다고 가정하고 두 번째 for문부터 디버깅한 결과입니다.

i	a[i]	a[i]%2	cnt	출력
0	1	1	0	홀수의 개수 : 3개
1	2	0	1	
2	3	1	2	
3	4	0	3	
4	5	1		
5				

[문제 5]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 외부 스키마, External Schema
- ② 개념 스키마, Conceptual Schema
- ③ 내부 스키마, Internal Schema

[문제 6]

- ① 3
- ② 5

[디버깅]

i	j	a[i][j]	출력
0	0	0	0 1 2 3 4
	1	1	
	2	2	
	3	3	
	4	4	
	5		
1	0	1	0 1 2 3 4 1 2 3 4 5
	1	2	
	2	3	
	3	4	
	4	5	
	5		
2	0	2	0 1 2 3 4 1 2 3 4 5 2 3 4 5 6
	1	3	
	2	4	
	3	5	
	4	6	
	5		
3			

[문제 7]

i % j == 0

※ 답안 작성 시 주의 사항 : C언어에서 사용하는 형식에 맞게 정확히 작성해야 합니다. 조건에서 '같다'는 표현을 'i % j = 0'으로 하지 않도록 주의하세요.

[디버깅]

i	j	i % j	출력
1	1	0	1의 약수 : 1
	2	1	
	3	1	
	4	1	
	5	1	
	6		
2	1	0	1의 약수 : 1 2의 약수 : 1 2
	2	0	
	3	2	
	4	2	
	5	2	
	6		
3	1	0	1의 약수 : 1 2의 약수 : 1 2 3의 약수 : 1 3
	2	1	
	3	0	
	4	3	
	5	3	
	6		
4	1	0	1의 약수 : 1 2의 약수 : 1 2 3의 약수 : 1 3 4의 약수 : 1 2 4
	2	0	
	3	1	
	4	0	
	5	4	
	6		
5	1	0	1의 약수 : 1 2의 약수 : 1 2 3의 약수 : 1 3 4의 약수 : 1 2 4 5의 약수 : 1 5
	2	1	
	3	2	
	4	1	
	5	0	
	6		
6			

[문제 8]

- ① ALTER
- ② ADD

※ 답안 작성 시 주의 사항 : 대·소문자를 구분하지 않습니다. 단 스펠링이 하나라도 틀렸을 경우 부분 점수 1도 없는 오답으로 처리된다는 것을 잊지 마세요.

[문제 9]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 포렌식, Forensics
- ② DDoS
- ③ 킬 스위치, kill switch
- ④ 제로 데이 공격, zero day attack
- ⑤ 업무 연속성 계획, BCP, Business Continuity Planning

※ 답안 작성 시 주의 사항 : 한글 또는 영문을 Full-name이나 약어로 쓰라는 지시사항이 없을 경우 한글이나 영문 약어로 쓰는 것이 유리합니다. 영문을 Full-name으로 풀어쓰다가 스펠링을 틀리면 오답으로 처리되니까요.

[문제 10]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① 지그비, zigbee
- ② 스토리지 전용 네트워크, SAN, storage area network
- ③ 클라우드 소싱, crowd sourcing
- ④ IPv6, Internet Protocol version 6
- ⑤ 커넥티드 카, connected car

[문제 11]

- ① σ
- ② π
- ③ $\triangleright \triangleleft$
- ④ \div
- ⑤ \cup
- ⑥ \cap
- ⑦ $-$
- ⑧ \times

[문제 12]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① GIS, Geographic Information System
- ② DHCP, Dynamic Host Configuration Protocol
- ③ Kernel
- ④ Bluetooth
- ⑤ AlphaGo

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

① 지리적인 자료를 수집·저장·분석·출력할 수 있는 컴퓨터 응용 시스템이며, 모든 사물들의 위치 정보를 제공하기 위해 위성을 이용한다. 이것을 통해 우리는 지도에서 사물을 확인하는 단계를 넘어서서, 인터넷과 위성과 같은 다양한 미디어를 이용하여 지리 데이터를 수집·구축·분석·처리함으로써 고품질의 공간 정보를 생산하는 것이 가능하다.

② TCP/IP 통신을 실행하기 위해 필요한 설정 정보를 자동적으로 할당하고 관리하기 위한 통신 규약이다. RFC 1541에서 규정하고 있으며, TCP/IP 환경의 통신망에서 IP 주소의 일률적인 관리 서비스를 제공한다. 이것을 제공하는 서버는 클라이언트의 요구에 따라 IP 주소를 할당한다.

③ 운영 체제에서 가장 핵심적인 역할을 한다. 메모리나 프로세서와 같은 자원을 관리하거나, 시스템이 원활히 운영되도록 제어하는 역할을 수행한다. 일반적으로 종료된 입·출력 연산과 같이 서비스에 대한 경쟁적인 모든 요청들을 처리하는 인터럽트 처리기, 프로그램들의 처리 시간을 어떤 순서대로 나눌지 결정하는 스케줄러, 그리고 각각의 프로세스에게 권한을 부여하는 관리자를 포함한다.

④ 에릭슨, IBM, 인텔, 노키아, 도시바에 의해서 1998년에 설립된 단체에서 나온 무선 개인 영역 네트워크 기술이다. 이것은 점 대 점 또는 다중 점 간의 응용 프로그램을 지원하는 데이터나 디지털 음성 등의 단거리 전송을 위한 공개된 표준이다. 처음에 나온 응용 프로그램들은 핸드프리 방식의 헤드폰이나 자동차 오디오 시스템에 연결하기 위해 휴대폰에 장착되었다.

⑤ 런던에 있는 Alphabet 사(社)의 구글 딥마인드에 의해 개발되었으며, 제로·마스터·리 등의 여러 가지 버전이 있다. 이것은 신경망 구조인 인간의 뇌와 같이 작동한다. 이 신경망은 정책망(Policy Network)과 가치망(Value Network)이라는 두 가지 신경망의 결합을 통해 만들어졌다. 정책망은 다음에 돌은 어디에 두는지 선택하는 알고리즘이고, 가치망은 승자를 예측하는 역할을 수행한다. 2015년 10월에는 핸디캡 없이 프로 바둑 기사를 이긴 첫 컴퓨터 프로그램으로 유명하다.

기출문제 정답

[문제 1]

- ① INPUT(2)
- ② INPUT(T+2)
- ③ BINARY(T+1)
- ④ INPUT(J+1)
- ⑤ GRAY(J+1)
- ⑥ K = 1, 4, 1 또는 K = 4, 1, -1

[디버깅]

순서도에 대해 INPUT 배열에 11101이 순서대로 입력된다고 가정하고, 디버깅한 결과입니다.

INPUT	T	BINARY	INPUT[T+2] = BINARY[T]	K	DECIMAL	출력
1	1				0	9
1 1	2			1	8	
1 1 1	3			2	8	
1 1 1 0	4			3	8	
1 1 1 0 1	5	1		4	9	
	1	1 0	Yes			
	2	1 0 0	Yes			
	3	1 0 0 1	No			

순서도에 대해 INPUT 배열에 01001이 순서대로 입력된다고 가정하고, 디버깅한 결과입니다.

INPUT	T	GRAY	J	INPUT[J+1] = INPUT[J+2]	L	출력
0	1				1	1
0 1	2				2	1
0 1 0	3				3	0
0 1 0 0	4				4	1
0 1 0 0 1	5	1				
		1 1	1	No		
		1 1 0	2	Yes		
		1 1 0 1	3	No		

[문제 2]

12

[디버깅]

a	b	c	sum	i	출력
1	1	2	2	3	12
1	2	3	4	4	
2	3	5	7	5	
3	5		12	6	

[문제 3]

- ① $n - 1$
- ② `hist[i]`

※ 답안 작성 시 주의 사항

배열의 요소를 지정할 때는 반드시 `hist[i]`와 같이 대괄호를 사용해야 합니다. `hist(i)`와 같이 소괄호를 사용하지 않도록 주의하세요.

[디버깅]

`rand()`는 난수를 발생시키는 함수이므로 코드를 실행할 때마다 `rand()` 함수로 산출되는 값과 `hist` 배열에 저장되는 값은 매번 달라집니다.

hist[]						rand()	rand()%6	n	i	출력
hist[0]	hist[1]	hist[2]	hist[3]	hist[4]	hist[5]					
0	0	0	0	0	0			0	0	
0	0	1	0	0	0	18560	2	3	1	
0	0	2	0	0	0	8056	2	3	2	
0	1	2	0	0	0	27193	1	2	3	[1] = 14
1	1	2	0	0	0	12816	0	1	4	[2] = 17
2	1	2	0	0	0	23814	0	1	5	[3] = 19
⋮						⋮	⋮	⋮	⋮	[4] = 14
13	16	17	14	18	18	16015	1	2	96	[5] = 18
13	16	18	14	18	18	29300	2	3	97	[6] = 18
14	16	18	14	18	18	5604	0	1	98	
14	17	18	14	18	18	30187	1	2	99	
14	17	19	14	18	18	8132	2	3	100	

※ `srand()` : 난수를 발생시킬 초기 값인 시드(seed)를 설정합니다. 매 초마다 난수를 발생시키는 경우에는 보통 현재 시간 값을 사용합니다.

※ `time(NULL)` : 1970년 1월 1일 이후부터 현재까지의 시간을 초 단위로 반환합니다.

[문제 4]

- ① `Next`
- ② `Pop()`

※ 답안 작성 시 주의 사항

C언어에서 사용하는 변수명은 대소문자를 구분하기 때문에 변수명을 작성할 때는 대소문자를 구분해서 정확히 작성해야 합니다.

[풀이]

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

struct NODE {
    int data;
    struct NODE *Next;
};

❶ struct NODE *head;
```

NODE 구조체를 정의합니다.
 정수형 변수 `data`를 선언합니다.
 NODE 구조체의 포인터 변수 `Next`를 선언합니다.

int data (4Byte)	struct NODE *Next (4Byte)
데이터를 저장할 멤버	다음 노드의 주소를 저장할 포인터

- ❶ NODE 자료형 포인터 변수 head를 전역 변수로 선언합니다. 포인터 변수의 크기는 4Byte이므로 메모리에 4Byte 크기의 공간이 할당됩니다.

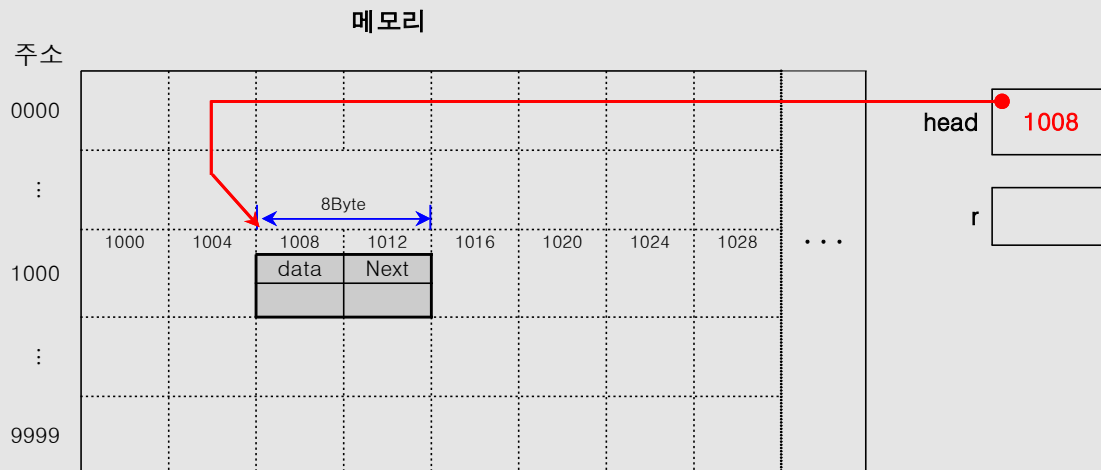
모든 C 프로그램은 반드시 main() 함수부터 시작해야 합니다.

```
main() {
❶ int r;

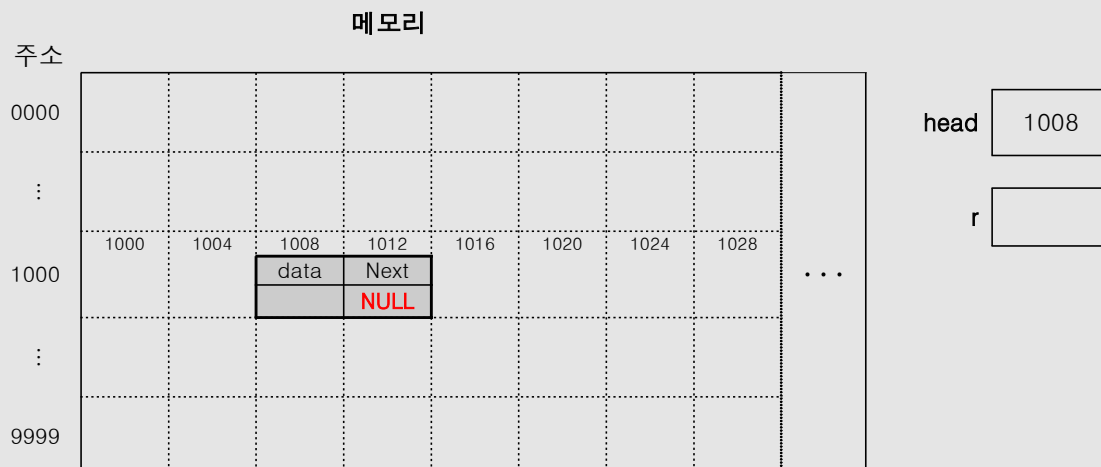
❷ head = malloc(sizeof(struct NODE));
❸ head->Next = NULL;

❹ Push(10);
   Push(20);
   Push(30);
}
```

- ❶ 정수형 변수 r을 선언합니다. 메모리의 어딘가에 4Byte 공간이 할당됩니다.
- ❷ malloc 함수가 메모리에서 NODE 구조체의 크기인 8Byte의 빈 영역 찾아 할당한 다음 그 영역의 시작 주소를 포인터 변수 head에 저장합니다. 이제 head는 할당된 공간의 시작 주소를 가리키는 것이죠. malloc 함수가 동적으로 할당하는 것이므로 여기서 지정한 주소 1008은 임의로 정한 것이며, 이해를 돕기 위해 주소를 10진수로 표현했습니다.



- ❸ head의 Next에 NULL을 저장합니다. head가 가리키고 있는 1008 번지는 NODE 구조체의 크기만큼 할당된 공간이므로 head의 Next는 1012 번지 이후의 4Byte를 의미합니다. 그곳에 NULL을 저장합니다. Next는 다음 노드의 주소를 지정하는 역할을 하고, NULL은 이론적으로 아무것도 없음을 의미합니다. 포인터 변수 head의 용도는 리스트 구조에서 첫 번째 노드의 주소를 저장하는 것입니다.



④ 10을 인수로 하여 Push 함수를 호출합니다.

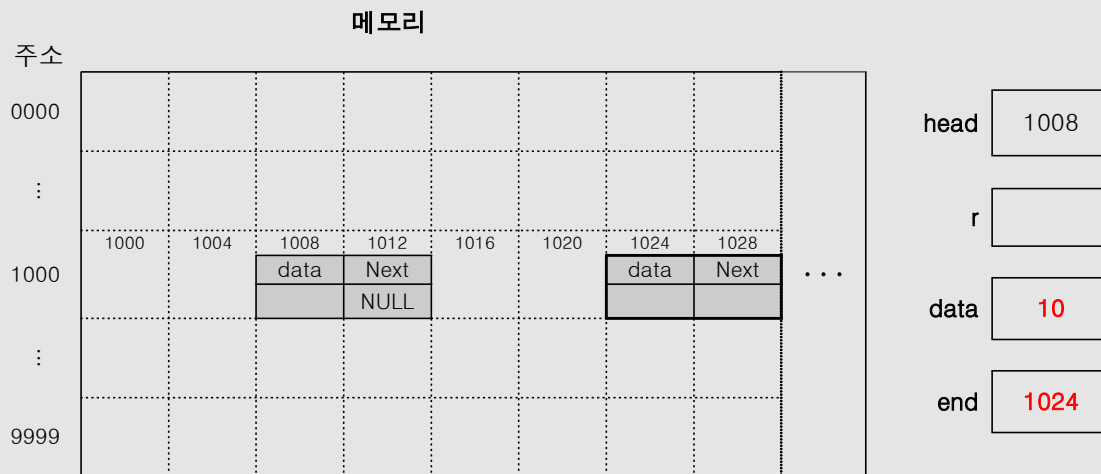
```

⑤ void Push(int data) {
⑥ struct NODE *end = malloc(sizeof(struct NODE));
⑦ end->Next = head->Next;
⑧ end->data = data;
⑨ head->Next = end;
⑩ }

```

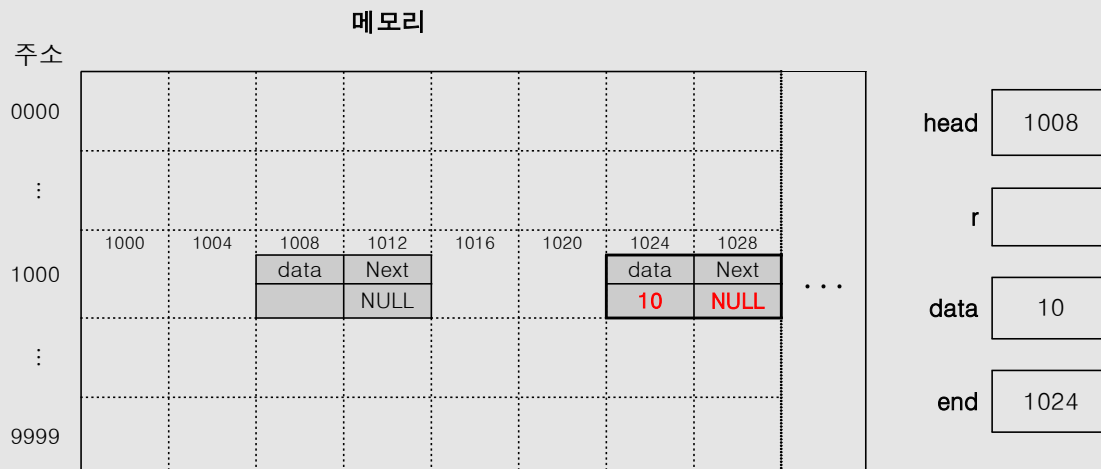
⑤ 함수의 리턴 값이 없으므로 void를 붙입니다. 정수형 변수 data를 지역 변수로 선언하고 ④번에서 보낸 10을 받아 그곳에 저장합니다.

⑥ NODE 자료형 포인터 변수 end를 지역 변수로 선언하고, NODE 구조체의 크기인 8Byte의 빈 영역을 찾아 할당한 다음 그 영역의 시작 주소인 1024를 포인터 변수 end에 저장합니다. 이제 end는 새로 할당된 공간의 시작 주소를 가리키는 것이죠. 포인터 변수 end의 용도는 리스트 구조에서 마지막 노드의 주소를 저장하는 것입니다.

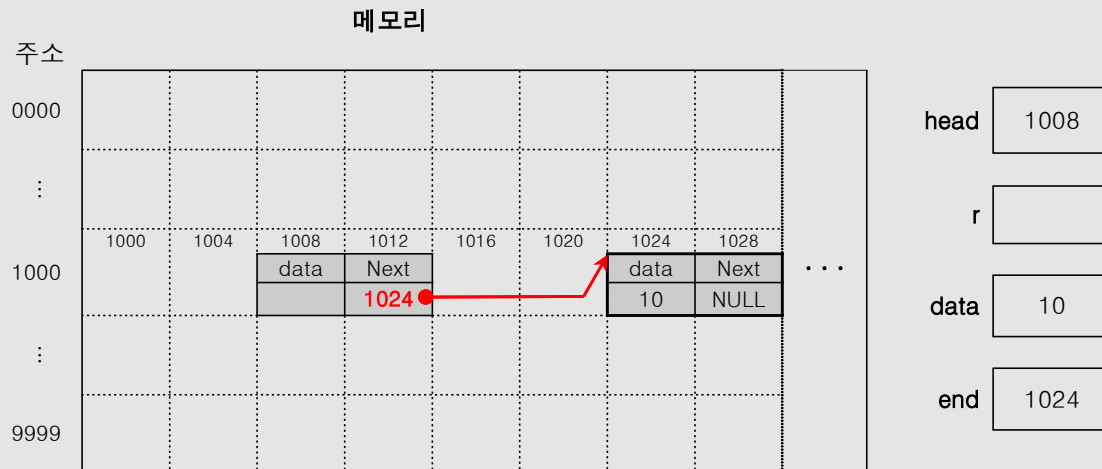


⑦ end의 Next에 head의 Next를 지정합니다. head의 Next가 NULL이므로 end의 Next에는 NULL이 저장됩니다. 지금 과정은 end가 리스트 구조에서 마지막 노드임을 지정하는 것으로 head의 Next가 다음 노드를 가리키도록 하기 위한 사전 작업입니다.

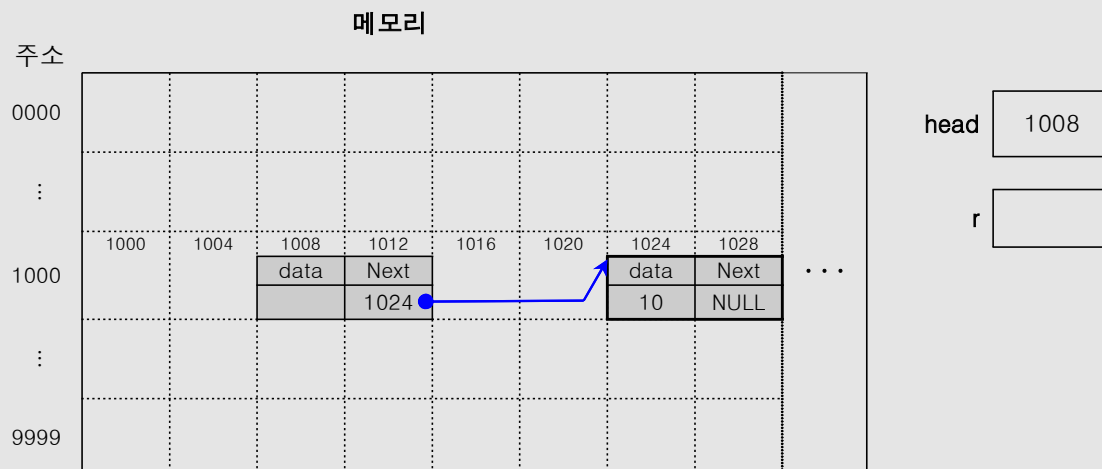
⑧ end의 data에 변수 data의 값 10을 저장합니다. end가 가리키고 있는 1024 번지는 NODE 구조체의 크기 만큼 할당된 공간이므로 end의 data란 1024 번지 이후의 4Byte를 의미합니다. 그곳에 변수 data의 값 10을 저장합니다.



- ⑨ head의 Next에 현재 end가 가지고 있는 값 1024를 저장합니다. 리스트 구조로 볼 때 이제 첫 번째 노드는 head가 가리키는 1008 번지부터 시작되는 8Byte이고 다음 노드는 head의 Next가 가리키고 있는 1024 번지부터 시작되는 8Byte입니다.



- ⑩ 함수를 마치고 Push 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Push 함수 안에서 선언된 지역 변수 end와 data는 메모리에서 제거됩니다.



```
main() {
    int r;

    head = malloc(sizeof(struct NODE));
    head->Next = NULL;

    Push(10);
    ⑪ Push(20);
    Push(30);
}
```

- ⑪ 20을 인수로 하여 Push 함수를 호출합니다. 번으로 이동합니다.

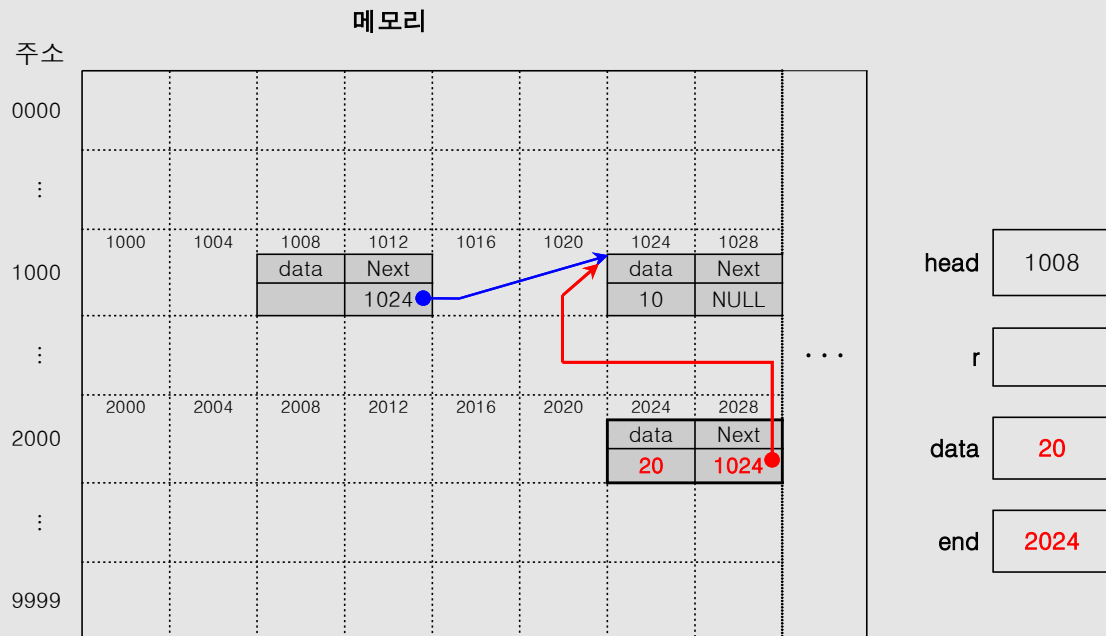
```
⑫ void Push(int data) {
    struct NODE *end = malloc(sizeof(struct NODE));
    end->Next = head->Next;
    end->data = data;
    head->Next = end;
}
```

⑫ 지역 변수 data에 뉘번에서 보낸 20이 저장됩니다.

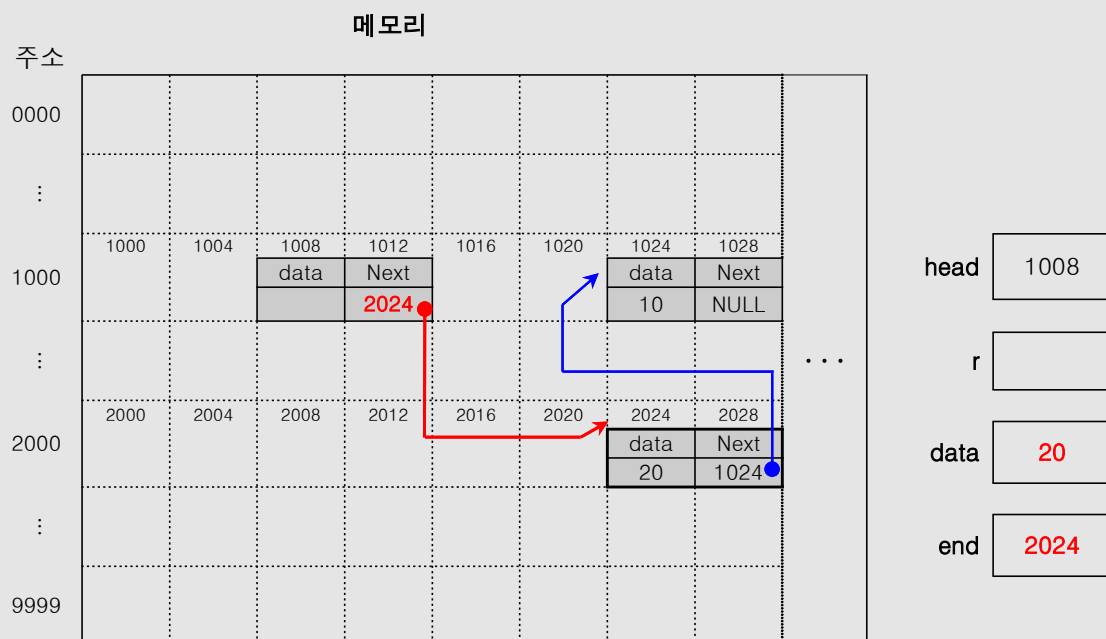
⑬ NODE 구조체의 크기인 8Byte의 빈 영역을 찾아 할당한 다음 그 영역의 시작 주소인 2024를 end에 저장합니다.

end의 Next에 head의 Next를 지정합니다. head의 Next가 1024이므로 end의 Next에는 1024가 저장됩니다.

end의 data에 변수 data의 값 20을 저장합니다.



head의 Next에 현재 end가 가지고 있는 값 2024를 저장합니다.



함수를 마치고 Push 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Push 함수 안에서 선언된 지역 변수 end와 data는 메모리에서 제거됩니다.

```

main() {
    int r;

    head = malloc(sizeof(struct NODE));
    head->Next = NULL;

    Push(10);
    Push(20);
    18 Push(30);
}

```

18 30을 인수로 하여 Push 함수를 호출합니다. 번으로 이동합니다.

```

19 void Push(int data) {
    struct NODE *end = malloc(sizeof(struct NODE));
    end->Next = head->Next;
    end->data = data;
    head->Next = end;
}

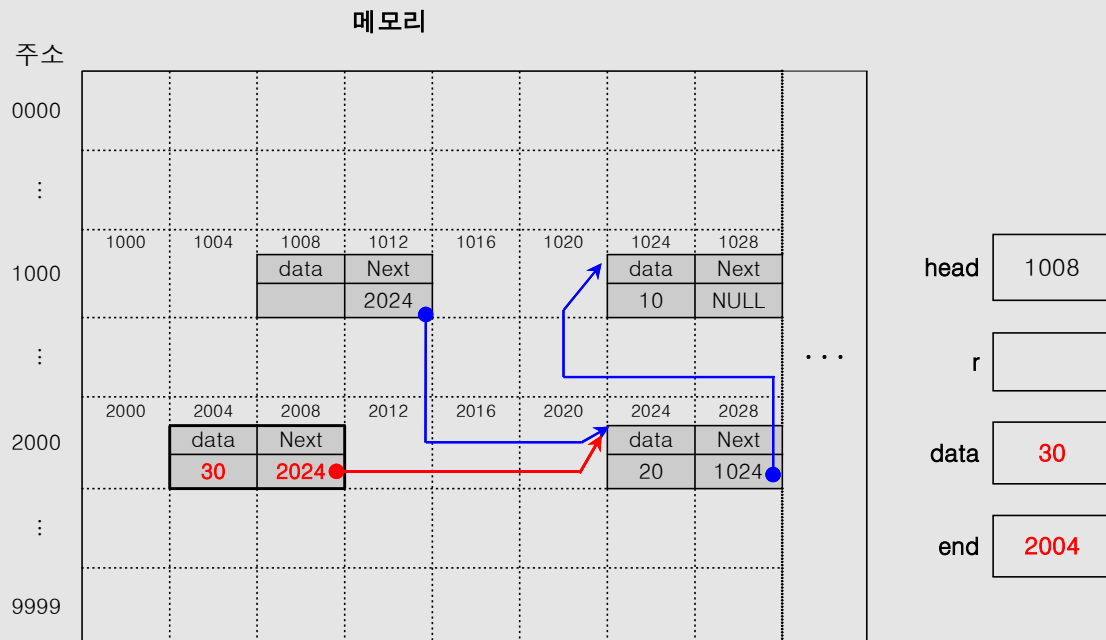
```

지역 변수 data에 이번에서 보낸 30이 저장됩니다.

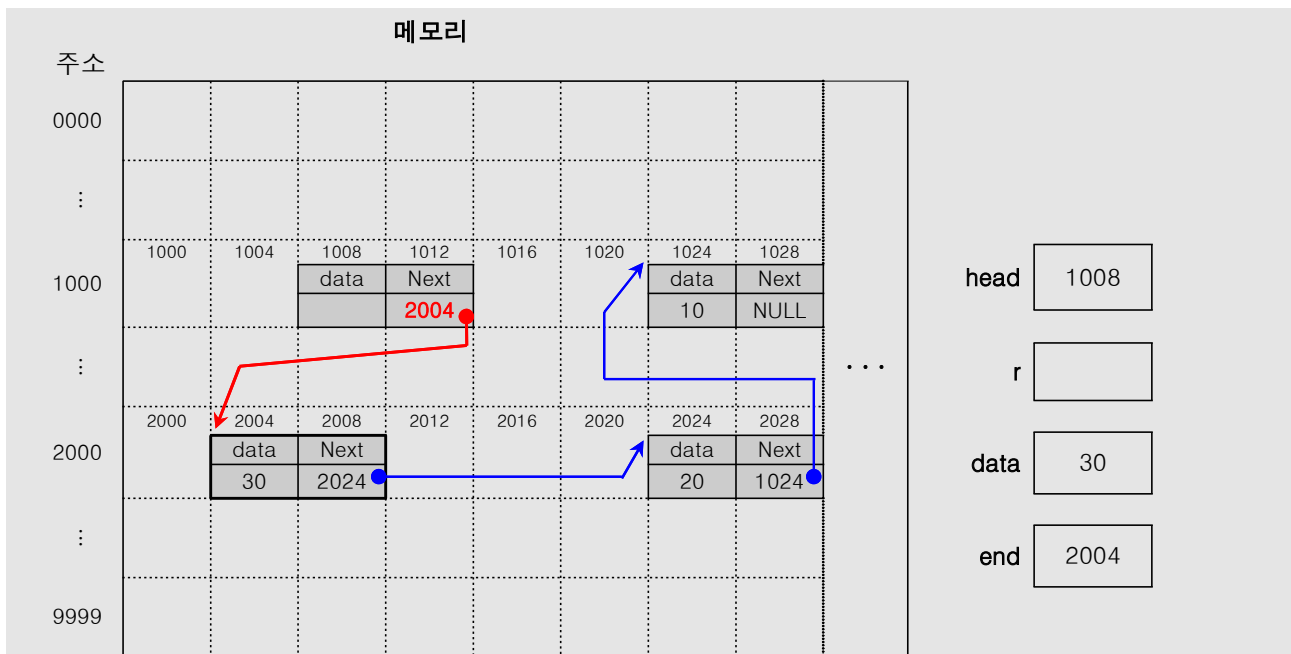
NODE 구조체의 크기인 8Byte의 빈 영역을 찾아 할당한 다음 그 영역의 시작 주소인 2004를 end에 저장합니다.

end의 Next에 head의 Next를 지정합니다. head의 Next가 2024이므로 end의 Next에는 2024가 저장됩니다.

end의 data에 변수 data의 값 30을 저장합니다.



head의 Next에 현재 end가 가지고 있는 값 2004를 저장합니다.



24 함수를 마치고 Push 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Push 함수 안에서 선언된 지역 변수 end와 data는 메모리에서 제거됩니다.

```

25 r = Pop();
   printf("%d\n", r);
   r = Pop();
   printf("%d\n", r);
   r = Pop();
   printf("%d\n", r);
}

```

인수 없이 Pop 함수를 호출한 다음 반환값을 r에 저장합니다. 번으로 이동합니다.

```

26 int Pop() {
    int a;

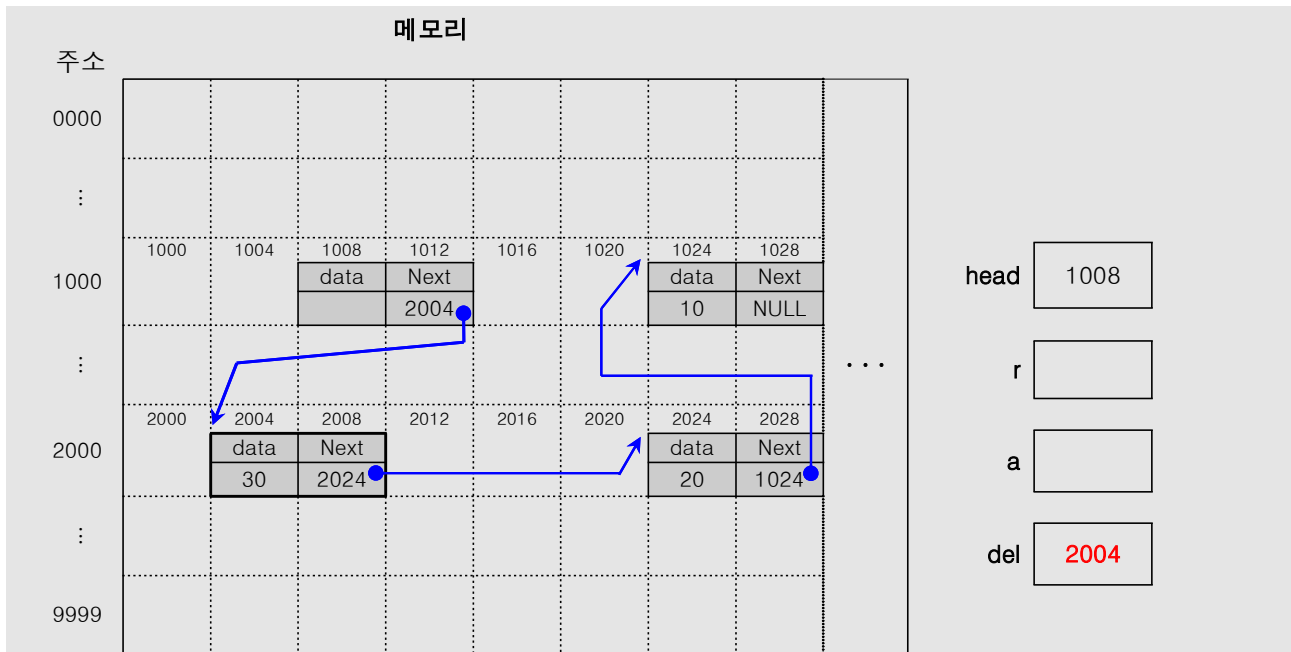
    struct NODE *del = head->Next;
    head->Next = del->Next;
    a = del->data;
    free(del);
    return a;
}

```

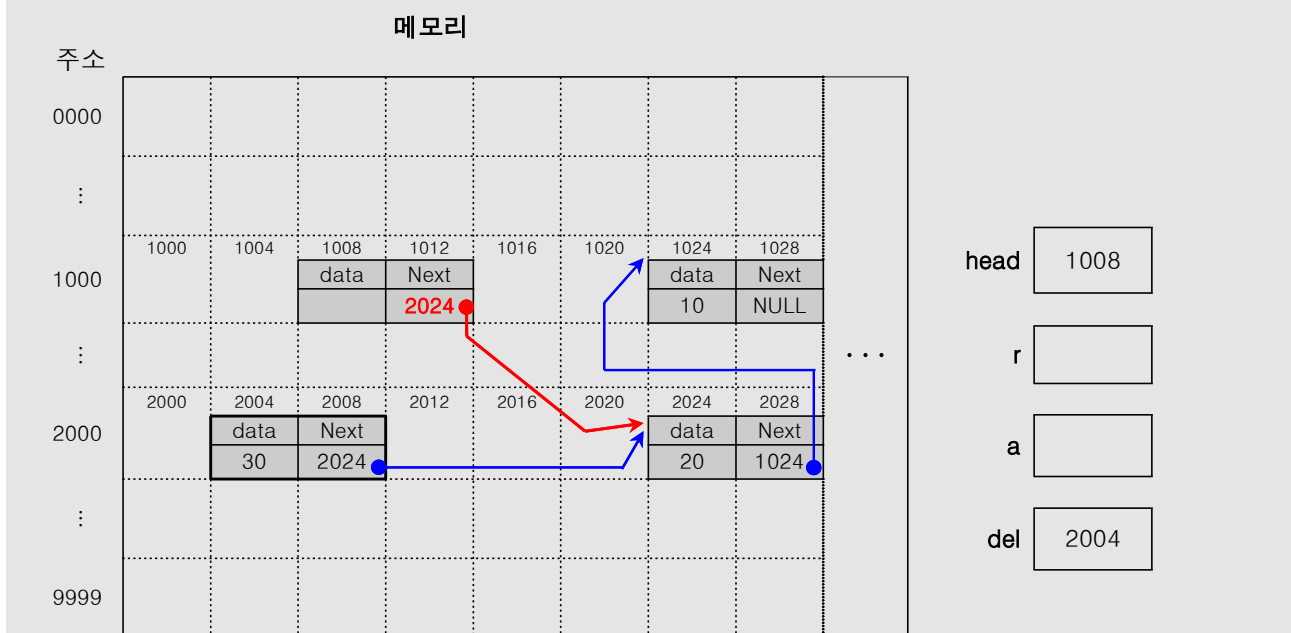
함수의 리턴 값이 있으므로 void를 생략합니다.

정수형 변수 a를 선언합니다. 메모리 어딘가에 4Byte 크기의 공간이 할당됩니다.

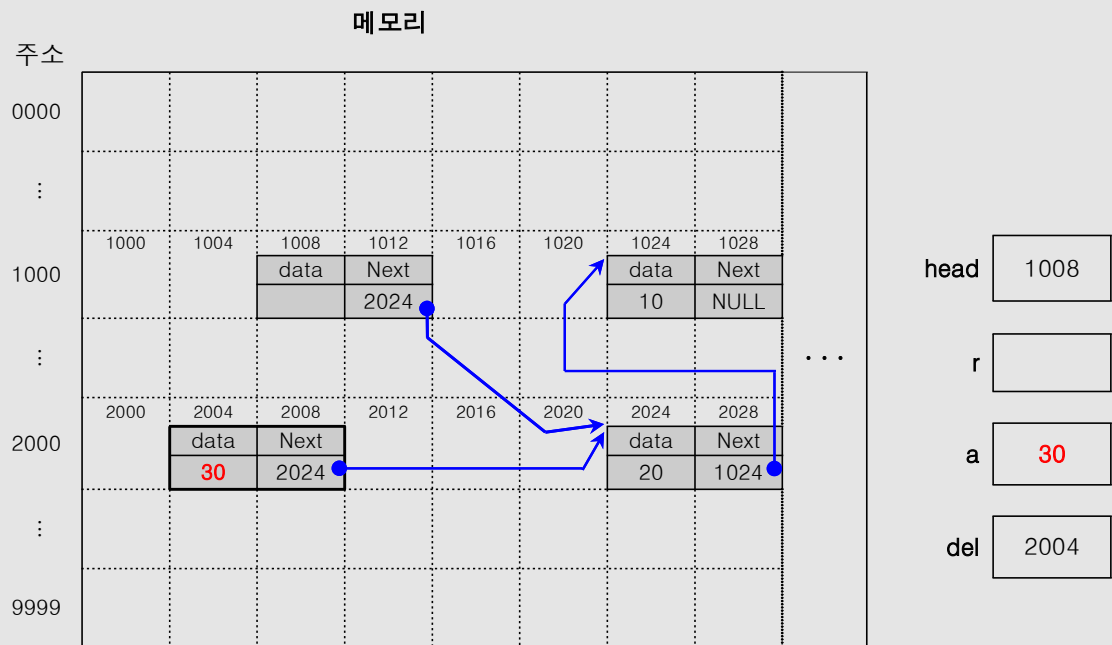
NODE 자료형 포인터 변수 del을 지역 변수로 선언하고, head의 Next인 2004를 저장합니다. 이제 del은 제거할 노드가 할당된 공간의 시작 주소인 2004를 가리키는 것이죠. 그러니까 포인터 변수 del의 용도는 리스트 구조에서 제거할 노드의 시작 주소를 저장하는 것입니다.



㉔ head의 Next에 del의 Next를 지정합니다. del의 Next가 2024이므로 head의 Next에는 2024가 저장됩니다. 2004 번지부터 시작되는 노드를 제거하고 난 후 그 노드가 가리키던 노드를 head의 Next가 가리키도록 하려는 것입니다. 그러니까 head의 Next는 항상 리스트 구조에서 첫 번째 노드를 가리키게 되는 것이죠.



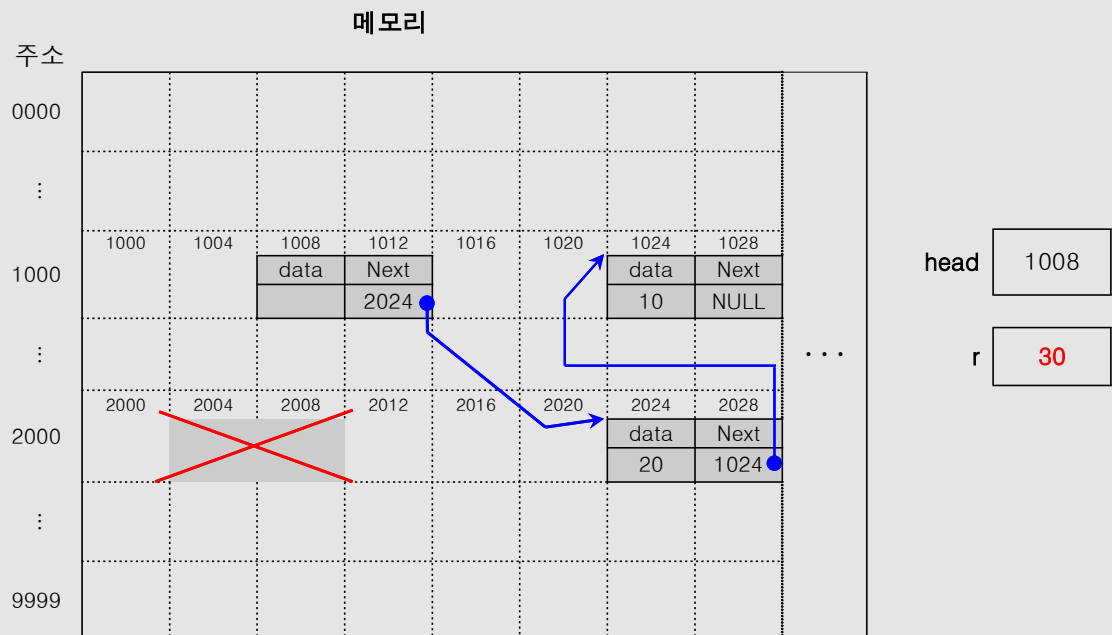
변수 a에 del의 data를 저장합니다. del이 가리키고 있는 2004 번지는 NODE 구조체의 크기만큼 할당된 공간이므로 del의 data란 2004 번지 이후의 4Byte를 의미합니다. 여기에 저장되어 있는 30을 변수 a에 저장합니다.



❶ del이 가리키고 있는 메모리 공간을 해제합니다. 즉 del이 가리키고 있는 2004 번지부터 할당된 8Byte 공간을 해제합니다.

※ free() : malloc() 함수에 의해 동적으로 할당된 메모리를 해제합니다.

30을 가지고 Pop 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Pop 함수 안에서 선언된 지역 변수 a와 del은 메모리에서 제거되고 변수 a의 값은 main() 함수의 변수 r로 전달됩니다.



```

r = Pop();
❶ printf("%d\n", r);
r = Pop();
printf("%d\n", r);
r = Pop();
printf("%d\n", r);
}

```

r의 값을 정수형으로 출력한 후 커서를 다음 줄 처음으로 옮깁니다.

결과 30

34 인수 없이 Pop 함수를 호출한 다음 결과를 r로 받습니다. 번으로 이동합니다.

```
35 int Pop() {
    int a;

    struct NODE *del = head->Next;
    head->Next = del->Next;
    a = del->data;
    free(del);
    return a;
}
```

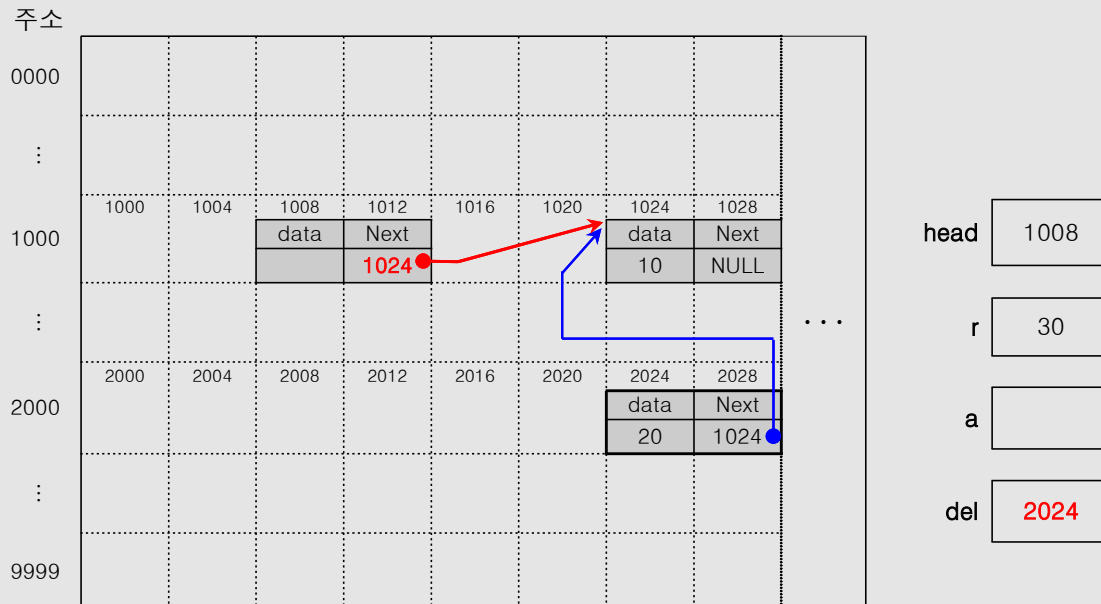
함수의 리턴 값이 있으므로 void를 생략합니다.

정수형 변수 a를 선언합니다.

NODE 자료형 포인터 변수 del을 지역 변수로 선언하고, head의 Next인 2024를 저장합니다.

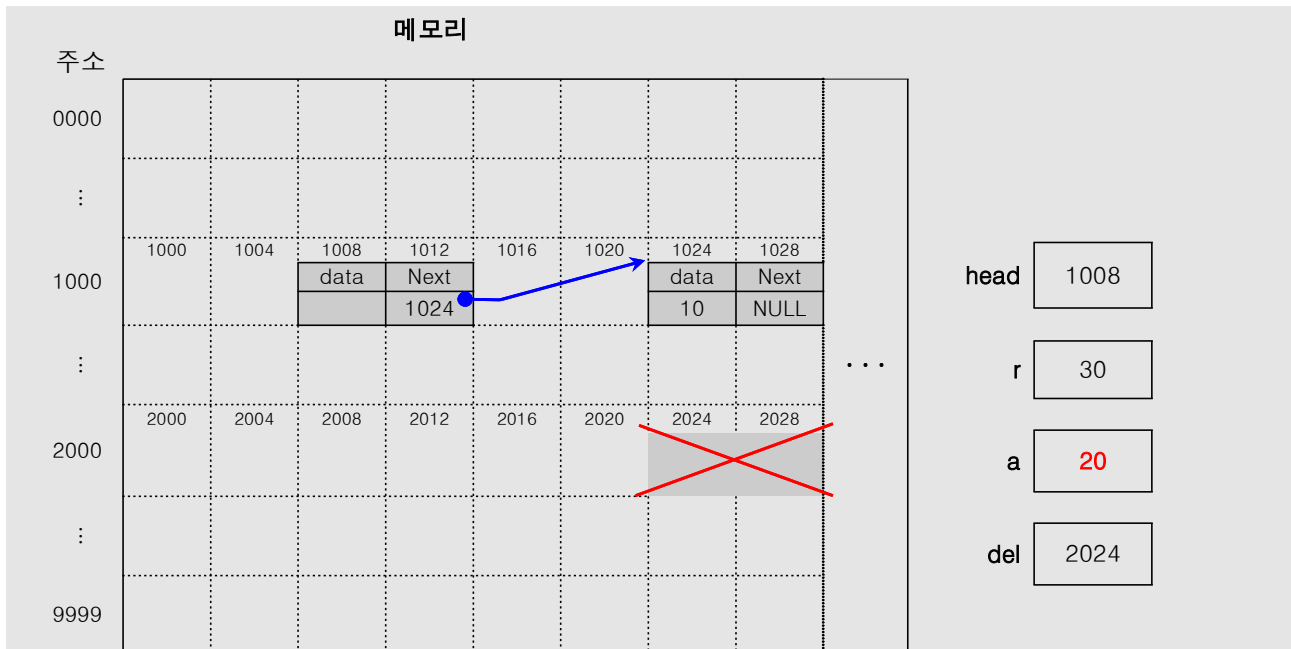
head의 Next에 del의 Next를 지정합니다.

메모리



변수 a에 del의 data, 즉 20을 저장합니다.

del이 가리키고 있는 메모리 공간을 해제합니다.



- ④1 20을 가지고 Pop 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Pop 함수 안에서 선언된 지역 변수 a와 del은 메모리에서 제거되고 변수 a의 값은 main() 함수의 변수 r로 전달됩니다.

```

r = Pop();
printf("%d\n", r);
r = Pop();
④2 printf("%d\n", r);
r = Pop();
printf("%d\n", r);
}

```

r의 값을 정수형으로 출력한 후 커서를 다음 줄 처음으로 옮깁니다.

결과 30
20

인수 없이 Pop 함수를 호출한 다음 결과를 r로 받습니다. 번으로 이동합니다.

```

④4 int Pop() {
    int a;

    struct NODE *del = head->Next;
    head->Next = del->Next;
    a = del->data;
    free(del);
    return a;
}

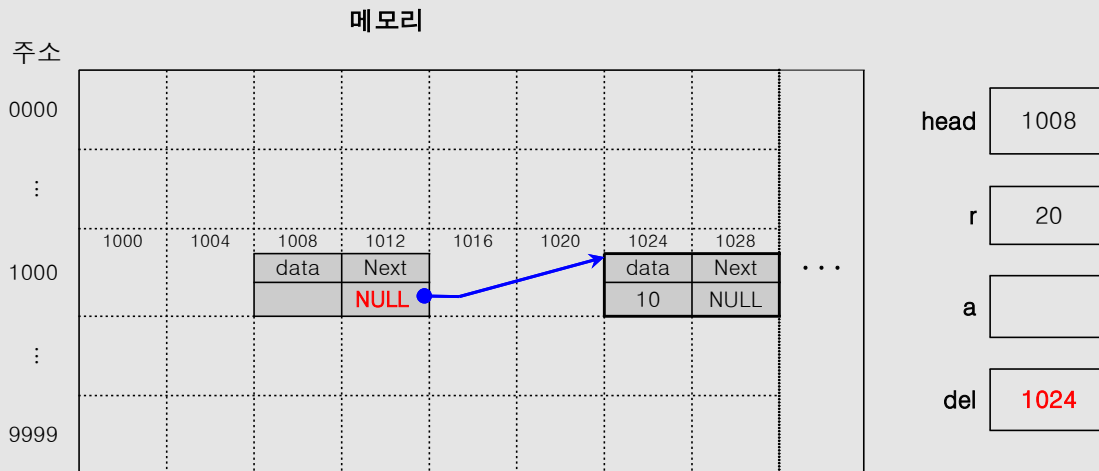
```

함수의 리턴 값이 있으므로 void를 생략합니다.

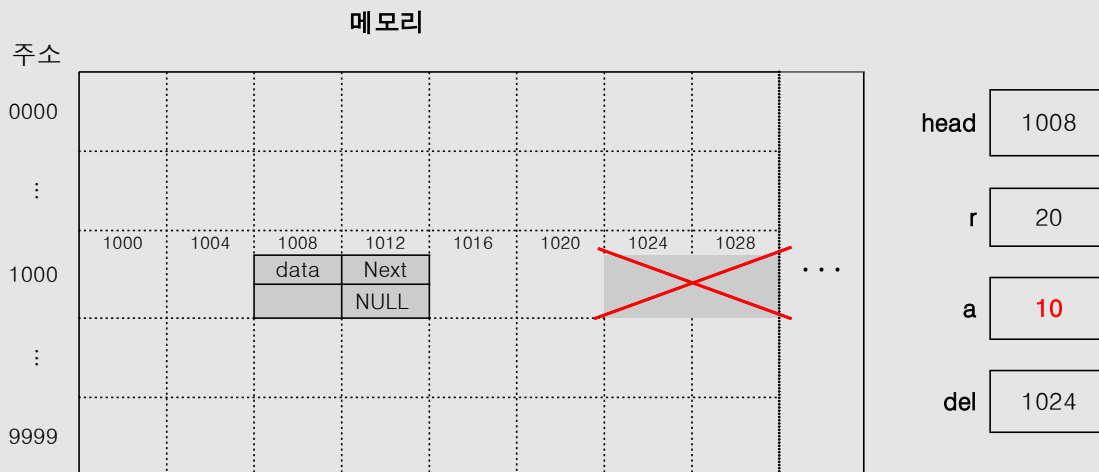
정수형 변수 a를 선언합니다.

NODE 자료형 포인터 변수 del을 지역 변수로 선언하고, head의 Next인 1024를 저장합니다.

head의 Next에 del의 Next를 지정합니다.



- ④8 변수 a에 del의 data, 즉 10을 저장합니다.
del이 가리키고 있는 메모리 공간을 해제합니다.



10을 가지고 Pop 함수를 호출했던 main() 함수로 제어를 옮깁니다. 이때 Pop 함수 안에서 선언된 지역 변수 a와 del은 메모리에서 제거되고 변수 a의 값은 main() 함수의 변수 r로 전달됩니다.

```

r = Pop();
printf("%d\n", r);
r = Pop();
printf("%d\n", r);
r = Pop();
⑤1 printf("%d\n", r);
}

```

r의 값을 정수형으로 출력한 후 커서를 다음 줄 처음으로 옮깁니다.

```

30
20
결과 10

```

프로그램을 종료합니다.

[문제 5]

(1) 1

(2) SELECT 학번 FROM 학생 WHERE 이름 LIKE '이%' ORDER BY 학년 DESC;

※ 답안 작성 시 주의 사항

대·소문자를 구분하지 않습니다. 단 스펠링이 하나라도 틀렸을 경우 부분 점수 1도 없는 오답으로 처리된다는 것을 잊지 마세요.

[풀이]

(1)

- SQL 실행 결과

count(*)
1

- SQL문

```
SELECT COUNT(*)   개수를 표시한다.
FROM 학생        <학생> 테이블을 대상으로 검색한다.
WHERE 학년=4;     '학년' 속성의 값이 4인 자료만을 대상으로 한다.
```

(2)

- SQL문

```
SELECT 학번        '학번'을 표시한다.
FROM 학생         <학생> 테이블을 대상으로 검색한다.
WHERE 이름 LIKE '이%'  '이름' 속성의 값이 '이'로 시작하는 자료만을 대상으로 한다.
ORDER BY 학년 DESC;   '학년' 속성의 값을 기준으로 내림차순 정렬한다.
```

- SQL 실행 결과

학번
14212054
16212060

[문제 6]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 애트리뷰트, Attribute, 컬럼, Column
- ② 튜플, Tuple, 로우, row
- ③ 도메인, Domain
- ④ 널, NULL
- ⑤ 차수, 디그리, Degree
- ⑥ 카디널리티, Cardinality, 기수, 대응수

[문제 7]

- ① O
- ② O
- ③ 다음 중 하나를 쓰면 됩니다.
타임 스탬프, Time Stamp

[문제 8]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 트러스트존, TrustZone
- ② 양자 암호 키 분배, QKD, Quantum Key Distribution

- ③ 지능형 지속 위협, APT, Advanced Persistent Threats
- ④ 가용성, Availability
- ⑤ 킬 스위치, Kill switch

※ 답안 작성 시 주의 사항

한글 또는 영문을 Full-name이나 약어로 쓰라는 지시사항이 없을 경우 한글이나 영문 약어로 쓰는 것이 유리합니다. 영문을 Full-name으로 풀어쓰다가 스펠링을 틀리면 오답으로 처리되니까요.

[문제 9]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 멤스, MEMS, Micro Electro Mechanical System
- ② 위키노믹스, Wikinomics
- ③ 웨바홀리즘, Webaholism
- ④ 컴패니언 스크린, Companion screen
- ⑤ IMT-2020, International Mobile Telecommunications-2020

[문제 10]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① EA, Enterprise Architecture
- ② 기업 자원 관리, ERP, Enterprise Resource Planning
- ③ 기업 애플리케이션 통합, 전사적 응용 통합, EAI, Enterprise Application Integration
- ④ 기술 경영, MOT, Management Of Technology
- ⑤ VOC, Voice Of Customer

[문제 11]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① VPN, Virtual Private Network
- ② VoIP, Voice over Internet Protocol
- ③ DRM, Digital Rights Management

※ 답안 작성 시 주의 사항

영문으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

(VPN)은 노드에 연결하기 위해 공공 회선을 사용함으로써 구성되는 망이다. 예를 들어, 데이터를 전송하기 위한 매체로서 인터넷을 사용한 망을 구성하도록 해주는 시스템이 많이 있다. 이러한 시스템은 승인된 사용자만 해당 망에 접근하도록 하고 데이터를 가로챌 수 없도록 하기 위하여 암호화 및 여타 보안 메커니즘을 사용한다.

(VoIP)는 인터넷(Internet)과 같은 인터넷 프로토콜(IP, Internet Protocol) 네트워크를 통해 음성 통신과 멀티미디어 세션을 구현하는 전달 기법, 통신 프로토콜, 방법론, 기술의 집합이다. 자주 접할 수 있는 용어 중 (VoIP)와 같은 뜻으로 종종 사용되는 것으로는 IP 전화, 인터넷 전화, 광대역 음성 통신, 브로드밴드 전화, 광대역 전화 등이 있다.

(DRM)은 데이터의 안전한 배포를 활성화하거나 불법 배포를 방지하여 인터넷이나 기타 디지털 매체를 통해 유통되는 데이터의 저작권을 보호하기 위한 시스템이다. (DRM) 시스템은 보통 데이터를 암호화하여 인증된 사용자만이 접속할 수 있게 하거나, 디지털워터마크의 사용 또는 이와 유사한 방식으로 콘텐츠를 작성하여 콘텐츠가 제한 없이 보급되지 않도록 하여 지적재산권을 보호한다.

기출문제 정답

[문제 1]

- ① J
- ② R
- ③ $SUM = SUM + J$
- ④ N
- ⑤ $LM + 1$

[디버깅]

LM	N	J	SUM	K	R	출력
0	4	1 2	0 1 3	2	0 0	6 28 496
	5	1 2	0 1	2	0 1	3
1	6	1 2 3	0 1 3 6	3	0 0 0	
	7	1 2 3	0 1	3	0 1 1	
	⋮	⋮	⋮	⋮	⋮	
2	28	1 2 3 4 ⋮ ⋮ 14	0 1 3 7 14 28	14	0 0 1 0 ⋮ ⋮ 0	
	⋮	⋮	⋮	⋮	⋮	
3	496	1 2 3 4 ⋮ ⋮ 246 247 248	0 1 3 7 15 31 62 128 248 496	248	0 0 1 0 ⋮ ⋮ 4 2 0	
	⋮	⋮	⋮	⋮	⋮	
	1000	1	0	500	0	

		2	1		0	
		.	3		.	
		.	.		.	
		.	.		.	
		500	.		0	
			1340			

[문제 2]

Sub Object

Super Object

※ 답안 작성 시 주의 사항

최근 출제되고 있는 프로그램 코드 문제는 이 문제처럼 코드에 대한 설명이 전혀 없거나 [문제 4]와 같이 간단한 예시만 제시되므로 수험생 스스로 코드를 파악한 후 문제에서 요구한 답을 찾아야 합니다. 그러므로 C나 JAVA 코드 문제를 풀 때는 실행 순서대로 디버깅을 수행하면서 문제 해결의 실마리를 찾는 습관을 들여야 합니다.

[해설]

```

① class SuperObject {
    public void paint() {
        draw();
    }

    ⑤ public void draw() {
        ⑥ draw();
        ⑨ System.out.println("Super Object");
    } ⑩
}

② class SubObject extends SuperObject{
    ③ public void paint() {
        ④ super.draw();
    } ⑪

    ⑦ public void draw() {
        ⑧ System.out.println("Sub Object");
    }
}

public class Test {
    public static void main(String[] args) {
        ① SuperObject a = new SubObject();
        ② a.paint();
    }
}

```

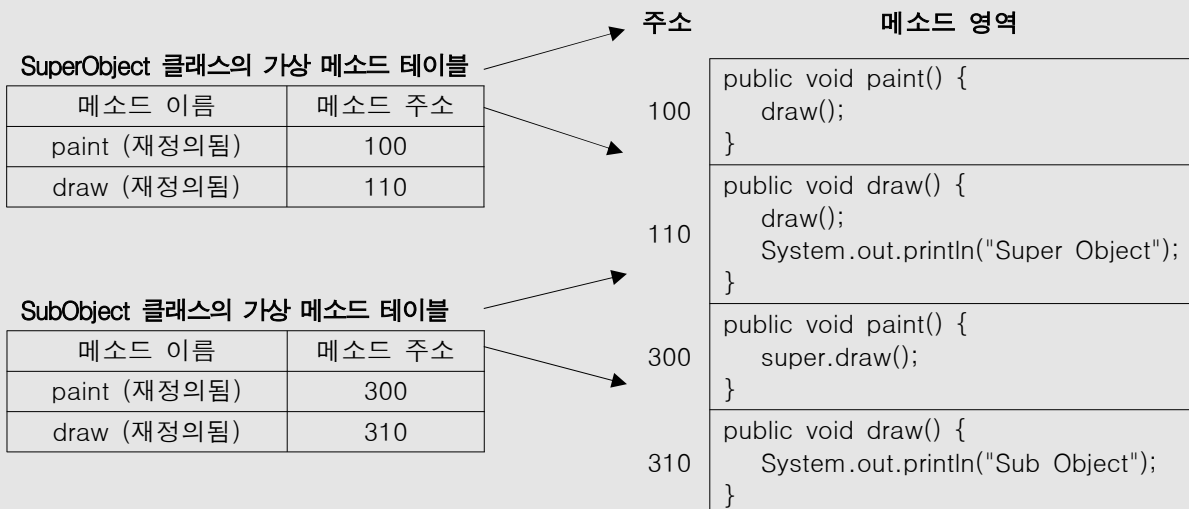
① 클래스 SuperObject를 정의한다.

② 클래스 SubObject를 정의하고 부모 클래스로 SuperObject를 지정하면서 SuperObject에 속한 변수와 메소

드를 상속받는다.

- **class A extends B** : A 클래스를 선언하면서 B 클래스에서 상속받음으로써 B 클래스의 변수 및 메소드를 사용할 수 있다.
- **메소드 재정의(오버라이딩)** : SubObject 클래스에 있는 paint(), draw() 메소드는 SuperObject 클래스에 있는 paint(), draw() 메소드와 이름은 같지만 메소드 안의 실행 코드는 다르다. 이와 같이 부모 클래스에서 정의한 메소드를 자식 클래스에서 다르게 정의해서 사용할 수 있는데, 이를 메소드 오버라이딩이라고 한다.

- ① • **[부모클래스명] [객체변수명] = new [자식클래스생성자()]** : 자식 클래스 생성자로 인스턴스를 생성할 때 자료형을 부모 클래스로 지정하면 생성된 인스턴스는 부모 클래스로 묵시적 클래스 형 변환이 된다. 이렇게 형 변환이 발생했을 때 부모 클래스와 자식 클래스에 같은 이름의 메소드가 존재하면 호출되는 메소드는 생성되는 인스턴스에 따라 결정된다. 즉 선언한 클래스 자료형이 아닌 생성된 자식 클래스 인스턴스의 메소드를 호출하는데, 이렇게 생성된 인스턴스의 메소드가 호출되는 기술을 가상 메소드라고 한다.
- 클래스에 속하는 멤버 변수는 인스턴스가 생성될 때마다 새로 생성되지만 메소드는 실행해야 할 코드의 집합이기 때문에 클래스의 인스턴스가 여러 개 생성된다고 해서 메소드가 여러 개 생성되지는 않는다.
 - 일반적으로 프로그램에서 메소드를 호출한다는 것은 그 메소드의 명령 집합이 있는 메모리 위치를 참조하여 명령을 실행하는 것인데, 가상 메소드의 경우에는 가상 메소드의 이름과 실제 메모리 주소가 짝을 이루는 '가상 메소드 테이블'이 만들어진다. 어떤 메소드가 호출되면 이 테이블에서 주소 값을 찾아 해당 메소드의 명령을 수행하는 것이다. 다음은 SuperObject 클래스와 SubObject 클래스의 가상 메소드 테이블이다.



- ② a.paint()는 a 객체의 자료형이 SuperObject이므로 SuperObject.paint()라고 생각할 수 있지만 ①에서 클래스 형 변환을 수행하였고 paint() 메소드가 자식 클래스에서 재정의된 메소드이므로 자식 클래스의 paint() 메소드가 수행된다. SubObject.paint()를 호출한다.
- ③ 리턴값과 매개변수가 없는 메소드 paint()의 시작점이다.
- ④ 부모 클래스를 호출하는 예약어 super를 사용했으므로 부모 클래스의 draw() 메소드를 수행한다. ⑤번으로 이동한다.
- ⑤ 리턴값과 매개변수가 없는 메소드 draw()의 시작점이다.
- ⑥ draw()를 수행한다. ①에서 클래스 형 변환을 수행하였고 draw() 메소드가 자식 클래스에서 재정의된 메소드이므로 자식 클래스의 draw() 메소드를 수행한다. ⑦번으로 이동한다.
- ⑦ 리턴값과 매개변수가 없는 메소드 draw()의 시작점이다.
- ⑧ 화면에 문자열 "Sub Object"를 출력하고, 다음 줄의 처음으로 커서를 이동시킨다. SubObject 클래스의 draw() 메소드가 종료되었으므로 메소드를 호출했던 ⑥번의 다음 줄인 ⑨번으로 이동한다.

결과 : Sub Object

⑨ 화면에 문자열 "Super Object"를 출력하고, 다음 줄의 처음으로 커서를 이동시킨다.

결과 : Sub Object
Super Object

⑩ SuperObject 클래스의 draw() 메소드가 종료되었으므로 메소드를 호출했던 ④번의 다음 줄로 이동한다.

⑪ SubObject 클래스의 paint() 메소드도 종료되었으므로 메소드를 호출했던 main() 메소드의 ⑤번으로 이동한 후 프로그램을 종료한다.

[문제 3]

27

[디버깅]

i	sum	i%4=0	출력
1	0	No	27
2		No	
3		No	
4	1	Yes	
5		No	
6		No	
7		No	
8	2	Yes	
9		No	
.	.	.	
.	.	.	
.	.	.	
104	26	Yes	
105		No	
106		No	
107		No	
108	27	Yes	
109		No	
110		No	
111			

[문제 4]

- ① input
- ② 10

[디버깅]

input에 5자리 숫자 12345가 입력되었다고 가정하고 디버깅한 결과입니다.

input	sum	input%10	출력
12345	0	5	15
1234	5	4	
123	9	3	
12	12	2	
1	14	1	
0	15		

[문제 5]

- ① GRANT
- ② REVOKE

[문제 6]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① Big Data, 빅데이터
- ② DSMS, 데이터 스트림 관리 시스템, Data Stream Management System
- ③ Anomaly, 이상
- ④ 부분 함수 종속, 부분 함수적 종속
- ⑤ 제 2정규형, 2NF
- ⑥ $A \rightarrow C$

※ 답안 작성 시 주의 사항

한글 또는 영문을 Full-name이나 약어로 쓰라는 지시사항이 없을 경우 한글이나 영문 약어로 쓰는 것이 유리합니다. 영문을 Full-name으로 풀어쓰다가 스펠링을 틀리면 오답으로 처리되니까요.

[문제 7]

- (1) 4
- (2) 3

[문제 8]

다음 중 한 가지를 쓰면 됩니다.

독립성, 격리성, Isolation

[문제 9]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 크라임웨어, Crimeware
- ② 해시 함수, Hash function
- ③ SSL, 보안 소켓 계층, Secure Sockets Layer
- ④ SET, 안전한 전자 거래, Secure Electronic Transaction
- ⑤ 루트킷, Rootkit

[문제 10]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① SaaS, 서비스형 소프트웨어, Software as a Service
- ② ITIL, 정보기술 인프라 라이브러리, Information Technology Infrastructure Library
- ③ RAID, 복수 배열 독립 디스크, Redundant Array of Independent Disks
- ④ 옵테인 메모리, Optane Memory
- ⑤ 온톨로지, Ontology

[문제 11]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① SEM, 검색 엔진 마케팅, Search Engine Marketing
- ② BSC, 균형 성과 기록표, Balanced Score Card
- ③ 기업인수자
- ④ 일반투자자

[문제 12]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① Interrupt
- ② PnP, Plug and Play
- ③ Register
- ④ Kernel
- ⑤ RAM, Random Access Memory

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

- ① 이 용어는 예상치 못한 이벤트가 발생했다는 것을 프로그램에 알리는 신호를 의미하며, 예외라고도 불린다. 이 신호는 프로그램이 이벤트를 처리하기 위해 일시적으로 멈추거나, 하고 있었던 작업으로 돌아갈 수 있도록 한다. 이 신호를 받으면 프로세서는 현재 수행중인 작업을 멈추고, 그 상태를 저장한 다음, 핸들러 또는 ISR이라고 불리는 이벤트 처리 기능을 실행하여 응답한다.
- ② 운영체제에서 새로 설치된 하드웨어를 자동으로 구성하는 능력이다. 운영체제가 연결되는 장치를 자동으로 인식해 적절한 구동 장치를 설치하므로 사용자는 구동 장치의 설정은 신경 쓰지 않고 장치를 연결하여 사용할 수 있다.
- ③ 처리될 데이터나 연산의 중간 결과를 저장하는 CPU 내부에 위치한 특별한 고속 저장 공간이다. 플립플롭(Flip-Flop)과 래치(Latch)들을 병렬로 결합한 것으로 가장 빠른 메모리이다.
- ④ 운영체제에서 가장 핵심적인 역할을 한다. 메모리나 프로세서와 같은 자원을 관리하거나, 시스템이 원활히 운영되도록 제어하는 역할을 수행한다. 일반적으로 종료된 입·출력 연산과 같이 서비스에 대한 경쟁적인 모든 요청들을 처리하는 인터럽트 처리기, 프로그램들의 처리 시간을 어떤 순서대로 나눌지 결정하는 스케줄러, 그리고 각각의 프로세스에게 권한을 부여하는 관리자를 포함한다.
- ⑤ 이 장치는 메모리 내부 데이터의 물리적 위치와는 관계없이 거의 동일한 시간 내에 데이터 항목들을 읽거나 쓸 수 있다. 데이터를 읽는 것만 가능한 ROM과 달리, 데이터를 읽고 쓰는 것이 가능하다. 또한 이것은 휘발성이라 이것에 저장된 어떤 데이터든 전원이 내려가면 곧바로 손실된다.

기출문제 정답

[문제 1]

- ① 10000
- ② M / 5
- ③ TM[K] + PM[K] 또는 PM[K] + TM[K]

[디버깅]

※ 다음과 같이 자료가 입력되었다고 가정하고 디버깅한 결과입니다.

<자 료>

임철식 537620

양동수 538973

이동훈 173105

QUIT

N	P	T	M	SW	K	배열	출력
임철식	537620	537620	10000	1	1	PM	출장비 지급 일람표 성명 출장비 만원 오전원 천원 오백원 백원 오십원 십원 오원 일원 임철식 537620 53 1 2 1 1 0 2 0 0
		7620	5000	0	2	53 1 2 1 1 0 2 0 0	
		2620	1000	1	3	TM	
		620	500	0	4	53 1 2 1 1 0 2 0 0	
		120	100	1	5	TM	
		20	50	0	6	53 1 2 1 1 0 2 0 0	
		20	10	1	7	TM	
		0	5	0	8	53 1 2 1 1 0 2 0 0	
		0	1	1	9	TM	
		0	0	0			
양동수	538973	538973	10000	1	1	PM	출장비 지급 일람표 성명 출장비 만원 오전원 천원 오백원 백원 오십원 십원 오원 일원 임철식 537620 53 1 2 1 1 0 2 0 0 양동수 538973 53 1 3 1 4 1 2 0 3
		8973	5000	0	2	53 1 3 1 4 1 2 0 3	
		3973	1000	1	3	TM	
		973	500	0	4	106 2 5 2 5 1 4 0 3	
		473	100	1	5	TM	
		73	50	0	6	106 2 5 2 5 1 4 0 3	
		23	10	1	7	TM	
		3	5	0	8	106 2 5 2 5 1 4 0 3	
		3	1	1	9	TM	
		0	0	0			
이동훈	173105	173105	10000	1	1	PM	출장비 지급 일람표 성명 출장비 만원 오전원 천원 오백원 백원 오십원 십원 오원 일원 임철식 537620 53 1 2 1 1 0 2 0 0 양동수 538973 53 1 3 1 4 1 2 0 3 이동훈 173105 17 0 3 0 1 0 0 1 0
		3105	5000	0	2	17 0 3 0 1 0 0 1 0	
		3105	1000	1	3	TM	
		105	500	0	4	123 2 8 2 6 1 4 1 3	
		105	100	1	5	TM	
		5	50	0	6	123 2 8 2 6 1 4 1 3	
		5	10	1	7	TM	
		5	5	0	8	123 2 8 2 6 1 4 1 3	
		0	1	1	9	TM	
		0	0	0			
QUIT					1	TM	출장비 지급 일람표 성명 출장비 만원 오전원 천원 오백원 백원 오십원 십원 오원 일원 임철식 537620 53 1 2 1 1 0 2 0 0 양동수 538973 53 1 3 1 4 1 2 0 3 이동훈 173105 17 0 3 0 1 0 0 1 0 전체화폐매수 123 2 8 2 6 1 4 1 3
					2	123 2 8 2 6 1 4 1 3	
					3	TM	
					4	123 2 8 2 6 1 4 1 3	
					5	TM	
					6	123 2 8 2 6 1 4 1 3	
					7	TM	
					8	123 2 8 2 6 1 4 1 3	
					9	TM	

[문제 2]

15

[디버깅]

i	numAry[i]	result	배열 numAry	출력
0	1	0	[0] [1] [2] [3] [4]	15
1	2	1	1	
2	3	3	1 2	
3	4	6	1 2 3	
4	5	10	1 2 3 4	
5		15	1 2 3 4 5	
1				
2				
3				
4				
5				

[향상된 for문 설명]

```
❶ for (int i : numAry)
    ❷ result += i;
```

❶ numAry 배열의 요수 수만큼 ❷번을 반복 수행한다.

• int i : numAry 배열의 각 요소가 할당될 변수를 선언한다.

• numAry : 배열의 이름을 적어준다. numAry 배열이 5개의 요소를 가지므로 각 요소를 i에 할당하면서 ❷번을 5회 수행한다.

[문제 3]

10

5

4

3

2

※ 답안 작성 시 주의 사항 : 프로그램의 실행 결과는 출력 양식을 정확히 지켜야 합니다. println을 통해 출력하였으므로 배열 n의 요소들을 출력할 때 마다 줄 나눔이 적용되어야 합니다.

[디버깅]

main 함수	
i	배열 n
	4 3 5 2 10
arr 함수로 이동	

arr 함수									출력										
배열 a	n	i	j	sw	a[j]	a[sw]	a[i]	temp											
<table><tr><td>4</td><td>3</td><td>5</td><td>2</td><td>10</td></tr></table>	4	3	5	2	10	5													
4	3	5	2	10															
<table><tr><td>4</td><td>3</td><td>5</td><td>2</td><td>10</td></tr><tr><td>10</td><td></td><td></td><td></td><td>4</td></tr></table>	4	3	5	2	10	10				4		0	1 2 3 4 5		0 2 4	3 5 2 10	4 5		
4	3	5	2	10															
10				4															
						4	10	4											
<table><tr><td>10</td><td>3</td><td>5</td><td>2</td><td>4</td></tr><tr><td></td><td>5</td><td>3</td><td></td><td></td></tr></table>	10	3	5	2	4		5	3				1	2 3 4 5		1 2	5 2 4	3 5		
10	3	5	2	4															
	5	3																	
						3	5	3											
<table><tr><td>10</td><td>5</td><td>3</td><td>2</td><td>4</td></tr><tr><td></td><td></td><td>4</td><td></td><td>3</td></tr></table>	10	5	3	2	4			4		3		2	3 4 5		2 4	2 4	3		
10	5	3	2	4															
		4		3															
						3	4	3											
<table><tr><td>10</td><td>5</td><td>4</td><td>2</td><td>3</td></tr><tr><td></td><td></td><td>3</td><td>2</td><td></td></tr></table>	10	5	4	2	3			3	2			3	4 5	3 4	3	2			
10	5	4	2	3															
		3	2																
						2	3	2											
		4																	
main 함수로 이동																			

main 함수							
배열 n	i	출력					
<table border="1"><tr><td>10</td><td>5</td><td>4</td><td>3</td><td>2</td></tr></table>	10	5	4	3	2	0	10
	10	5	4	3	2		
	1	5					
	2	4					
	3	3					
	4	2					
5							

[문제 4]

ch == '0'

※ 답안 작성 시 주의 사항 : 프로그램의 실행 결과는 출력 양식을 정확히 지켜야 합니다. 서식 지정자 '%s' 로 출력하였으므로 배열 str의 모든 요소를 출력해야 합니다.

[디버깅]

i	j	ch	str[i]	str[j]	배열 str	출력
0		1				
1		2			1 2 3 4 5 0 0 0	
2		3				
3		4				
4		5				
5		0				
4	0	1	1	5	1 2 3 4 5 0 0 0 5 1	
3						
2	1	2	2	4	5 2 3 4 1 0 0 0 4 2	
	2					54321000

[문제 5]

1 또는 1개

[해설]

- 대체키(Alternate Key)는 후보키가 둘 이상일 때 기본키를 제외한 나머지 후보키들을 말합니다.
- 후보키는 릴레이션을 구성하는 속성들 중에서 튜플을 유일하게 식별하기 위해 사용하는 속성들의 부분집합, 즉 기본키로 대체해서 사용할 수 있는 속성들을 말합니다.

<회원>

ID	이름	소속	지역	전화번호
2009001	허진욱	테니스	수원	010-5146-7812
2009112	김춘삼	테니스	서울	010-9711-1845
2010323	신민주	탁구	인천	010-4848-7915
2017794	임격정	야구	성남	010-5348-8776
2018655	김형규	볼링	서울	NULL

- ID : 기본키이므로 대체키가 될 수 없습니다.
- 이름 : 속성 값 중 중복된 값이 없으므로 대체키에 해당합니다.
- 소속 : 속성 값 중 중복된 값(테니스)이 있어 대체키가 될 수 없습니다.
- 지역 : 속성 값 중 중복된 값(서울)이 있어 대체키가 될 수 없습니다.
- 전화번호 : 기본키를 대체할 수 있는 대체키의 특성상 기본키와 마찬가지로 NULL 값을 가질 수 없어야 하는데 전화번호에는 NULL이 있으므로 대체키가 될 수 없습니다.

[문제 6]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① DDL 컴파일러, DDL 처리기, DDL 번역기
- ② 트랜잭션 관리자

[문제 7]

200, 100, 300

※ SQL문은 3개인데, 답안은 1개입니다. 답안 작성에 대한 별도의 조건이 제시되지 않았으므로 각 SQL문의 결과를 구분할 수 있도록 작성하면 됩니다.

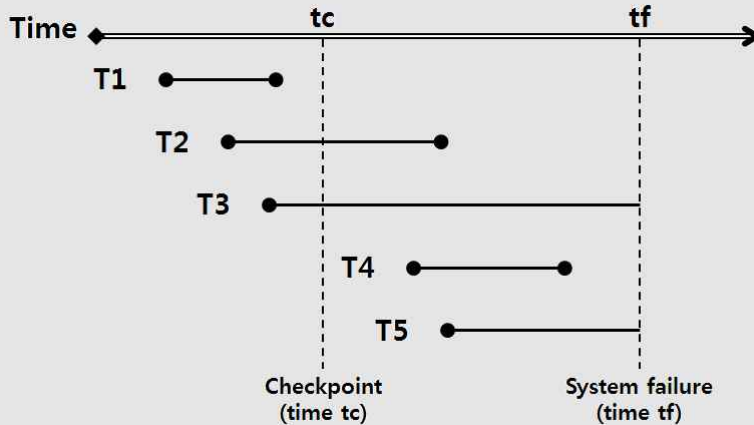
[문제 8]

INSERT INTO 학생 VALUES(193739, '홍길동', 3, '경영학개론', '010-1234-1234');

[문제 9]

- REDO : T2, T4
- UNDO : T3, T5

[해설]



그림에 Checkpoint가 있는 것으로 보아 회복 기법으로 검사점(Check Point) 기법이 적용되었음을 알 수 있습니다. 검사점 기법은 시스템에 장애가 발생하였을 경우, REDO와 UNDO 수행을 위해 로그 전체를 조사해야 하는 경우를 피하기 위한 기법으로, 시스템에 장애가 발생하면 가장 최근에 설정된 검사점에서부터 회복 작업을 수행하여 회복 시간을 단축시킵니다.

- **REDO** : 데이터베이스가 비정상적으로 종료되었을 때 디스크에 저장된 로그를 분석하여 트랜잭션의 시작(start)과 완료(commit)에 대한 기록이 있는 트랜잭션들의 작업을 재작업한다.
- **UNDO** : 데이터베이스가 비정상적으로 종료되었을 때 디스크에 저장된 로그를 분석하여 트랜잭션의 시작을 나타내는 'start'는 있지만 완료를 나타내는 'commit' 기록이 없는 트랜잭션들이 작업한 변경 내용들을 모두 취소한다.

시스템 장애가 발생한 시점에서 가장 최근의 Checkpoint(tc)를 기준으로 시작과 완료가 있는 T2, T4는 REDO 대상이 되고, 시작은 있지만 완료가 없는 T3, T5는 UNDO 대상이 됩니다. T1은 Checkpoint 이전에 완료된 작업이므로 검사점 기법이 적용된 이 문제에서는 REDO나 UNDO 대상에서 제외됩니다.

[문제 10]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① 스텍스넷, Stuxnet
- ② 다크 데이터, Dark Data
- ③ 킬 스위치, Kill Switch
- ④ APT, 지능형 지속 위협, Advanced Persistent Threats
- ⑤ 가용성, Availability

[문제 11]

각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① 오픈스택, OpenStack
- ② 블록체인, Blockchain
- ③ QoS, Quality of Service, 서비스 품질
- ④ 지그비, Zigbee

- ⑤ 멤스, MEMS, Micro Electro Mechanical System

[문제 12]

- ① EDI
- ② 데이터마이닝
- ③ 주문접수

[문제 13]

각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① DRM, Digital Rights Management
- ② VPN, Virtual Private Network
- ③ LBS, Location Based Service
- ④ DDoS, Distributed Denial of Service
- ⑤ VoIP, Voice over Internet Protocol

※ 답안 작성 시 주의 사항 : 영문으로 쓰라는 지시사항이 있을 경우 스펠링에 유의하세요. 스펠링을 한 개라도 잘못 쓰면 오답으로 처리되니까요.

[원문해석]

- ① 이것은 데이터의 안전한 보급을 활성화하거나 불법 보급을 방지하여 인터넷이나 기타 디지털 매체를 통해 유통되는 데이터의 저작권을 보호하기 위한 시스템이다. 일반적으로 이 시스템은 데이터를 암호화하여 허가된 사용자만이 접속할 수 있도록 하거나, 콘텐츠에 디지털 워터마크를 표시하거나, 또는 비슷한 방법을 이용하여 콘텐츠가 자유롭게 배포되는 것을 막아 지적 재산을 보호한다.
- ② 이것은 공공 회선을 사용해 노드를 연결하는 것으로 구축되는 네트워크이다. 인터넷을 데이터 전송 매체로 사용하는 네트워크를 생성할 수 있도록 하는 시스템이 다수 존재한다는 것을 예로 들 수 있다. 이러한 시스템은 인증된 사용자만이 네트워크에 접속할 수 있도록 하고 데이터가 도용될 수 없도록 하기 위해 암호화 및 기타 보안 메커니즘을 사용한다.
- ③ 이것은 기능을 제어하기 위해 위치 데이터를 사용하는 일반적인 수준의 프로그램 단계 서비스들이다. 오늘날 대부분의 이것들은 정보제공 서비스, 또는 오락 서비스와 같은 소셜 네트워크 서비스에서 많이 이용되곤 한다. 또한 모바일 네트워크를 통하여 모바일로 접속할 수 있으며, 모바일 기기의 지리적 위치에 대한 정보를 사용한다. 스마트폰과 태블릿 시장의 확장에도 이것은 점점 더 중요해 지고 있다.
- ④ 이것은 단일 시스템을 표적으로 훼손된(보통 트로이 목마에 감염된) 여러 시스템을 사용해 서비스 거부 공격을 유발하는 공격법이다. 표적이 된 시스템 및 해커의 제어 하에 분산 공격에 부당하게 사용된 모든 시스템이 이 공격의 피해자가 된다.
- ⑤ 이것은 전통적인 PSTN 회선 전송망이 아니라 IP를 사용해 음성 데이터를 패킷 형태로 송신하는 것을 통해 사용자가 인터넷을 전화 통화용 송신매체로 사용할 수 있도록 하는 하드웨어 및 소프트웨어 분류를 말한다. 이것의 장점 중 하나는 인터넷을 통해 개인 이메일을 발송하는 것과 마찬가지로, 사용자가 인터넷 접속비로 지불하는 비용 외에 인터넷을 사용한 전화 통화로 추가 비용이 발생하지 않는다는 것이다.



2020년 1회 정보처리기사 실기 정답 및 해설

[답안 작성 방법 안내]

'운영체제(OS; Operation System)'처럼 한글과 영문으로 제시되어 있는 경우 '운영체제', 'OS', 'Operation System' 중 1가지만 쓰면 됩니다.

[문제 1]

XML(eXtensible Markup Language)

[문제 2]

JSON(JavaScript Object Notation)

[문제 3]

머릿말(Header)

[문제 4]

0 1 2 3

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 콤마를 넣어 0, 1, 2, 3으로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

해설

모든 Java 프로그램은 반드시 main() 메소드부터 시작해야 한다.

```
public static void main(String[] args) {  
    ❶ int a[] = arr();  
        for(int i = 0; i < a.length; i++)  
            System.out.print(a[i] + " ");  
}
```

❶ 정수형 배열 a를 선언하고, 인수 없이 arr() 메소드를 호출한 다음 돌려받은 값을 배열 a에 저장한다.

```
    ❷ static int[] arr() {  
        ❸ int a[] = new int[4];  
        ❹ int b = a.length;  
        ❺ for(int i = 0; i < b; i++)  
            ❻ a[i] = i;  
        ❼ return a;  
    }
```

❷ 메소드의 리턴값이 정수형 배열인 arr() 메소드의 시작점이다.

❸ 4개의 요소를 갖는 정수형 배열 a를 선언한다.

❹ 정수형 변수 b를 선언하고 배열 a의 길이인 4로 초기화한다.

• length : length는 배열 클래스의 속성으로 배열 요소의 개수가 저장되어 있다. a 배열은 4개의 요소를 가지므로 a.length는 4를 가지고 있다.

❺ 반복 변수 i가 0에서 시작하여 1씩 증가하면서 b보다 작은 동안 ❻번 문장을 반복 수행한다.



⑥ a[i]에 반복 변수 i의 값을 저장한다.

	a[0]	a[1]	a[2]	a[3]
a[]	0	1	2	3

⑦ 배열 a에 저장된 값들을 호출한 곳(main() 메소드)으로 반환한다.

```
public static void main(String[] args) {
    ⑧ int a[] = arr();
    ⑨ for(int i = 0; i < a.length; i++)
    ⑩ System.out.print(a[i] + " ");
}
```

⑧ arr() 메소드로부터 반환받은 값들을 main() 메소드의 배열 a에 저장한다.

⑨ 반복 변수 i가 0에서 시작하여 1씩 증가하면서 a 배열의 길이 4보다 작은 동안 ⑩번 문장을 반복 수행한다.

⑩ a[i]의 값과 공백 한 칸을 출력한다.

[문제 5]

① 결합도(Coupling) ② 응집도(Cohesion)

[문제 6]

• 계산과정 : $(30,000 / 300) / 5 = 20$

• 답 : 20개월

해설

LOC 기법에서 개발 기간은 '노력(인원) / 투입 인원'이고, 노력(인원)은 'LOC(총 라인 수) / 1인당 월평균 생산 코드 라인 수'이므로 'LOC / 1인당 월평균 생산 코드 라인 수 / 투입 인원'에 값을 대입하여 답을 구할 수 있습니다.

[문제 7]

① 원자성(Atomicity) ② 독립성(Isolation)

[문제 8]

① 처리량(Throughput) ② 응답 시간(Response Time) ③ 경과 시간(Turn Around Time)

[문제 9]

다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

비정규화란 정규화된 데이터 모델을 통합, 중복, 분리하는 과정으로, 의도적으로 정규화 원칙을 위배하는 행위이다.

[문제 10]

MD5(Message-Digest algorithm 5)

[문제 11]

구문(Syntax), 의미(Semantics), 시간(Timing)

[문제 12]

LAND Attack(Local Area Network Denial Attack)



[답안 작성 방법 안내]

'운영체제(OS; Operation System)'처럼 한글과 영문으로 제시되어 있는 경우 '운영체제', 'OS', 'Operation System' 중 1가지만 쓰면 됩니다.

[문제 13]

다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

살충제 패러독스는 동일한 테스트 케이스로 동일한 테스트를 반복하면 더 이상 결함이 발견되지 않는 현상을 의미한다.

[문제 14]

-8

해설

```
#include <stdio.h>
main() {
  ① int c = 1;
  ② switch (3) {
    case 1: c += 3;
    case 2: c++;
  ③ case 3: c = 0;
  ④ case 4: c += 3;
  ⑤ case 5: c -= 10;
  ⑥ default: c--;
  }
  ⑦ printf("%d", c);
}
```

모든 case문에 break문이 생략되었으므로, switch문의 인수와 일치하는 'case 3' 문장부터 switch문이 종료될 때까지 모든 문장이 실행된다.

- ① 정수형 변수 c를 선언하고 1로 초기화한다. → c = 1
- ② 3에 해당하는 숫자를 찾아간다. 'case 3' 문장으로 이동한다.
- ③ c의 값을 0으로 치환한다. → c = 0
- ④ 'c = c + 3'과 동일하다. c의 값에 3을 더한다. → c = 3
- ⑤ 'c = c - 10'과 동일하다. c의 값에서 10을 뺀다. → c = -7
- ⑥ 'c = c - 1'과 동일하다. c의 값에서 1을 뺀다. → c = -8
- ⑦ c의 값을 출력한다. 결과 -8

[문제 15]

물리 계층(Physical Layer)

[문제 16]

F, H

해설

팬인(Fan-in)은 모듈로 들어오는(n) 선(Line)의 수라고 생각하면 됩니다. 그러므로 2개의 선이 들어오는 F, H 모듈이 답이 됩니다.



[문제 17]

① 200 ② 3 ③ 1

해설

① <SQL문>

```
SELECT DEPT
FROM STUDENT;
```

'DEPT'를 표시한다.
<STUDENT> 테이블을 대상으로 검색한다.

※ <STUDENT> 테이블에서 'DEPT'를 검색합니다. 총 200개의 튜플이 들어 있고 검색 조건이 없으므로 튜플의 수는 200개 입니다.

② <SQL문>

```
SELECT DISTINCT DEPT
FROM STUDENT;
```

'DEPT'를 표시하되, 같은 'DEPT' 속성의 값은 한 번만 표시한다.
<STUDENT> 테이블을 대상으로 검색한다.

※ <STUDENT> 테이블에서 'DEPT'를 검색하는 데 중복된 결과는 처음의 한 개만 검색에 포함시킵니다. 전기과 50개 튜플의 'DEPT' 속성의 값이 같으므로 1개, 전산과 100개 튜플의 'DEPT' 속성의 값이 같으므로 1개, 전자과 50개 튜플의 'DEPT' 속성의 값이 같으므로 1개를 검색에 포함시키므로 3개의 튜플이 검색됩니다.

③ <SQL문>

```
SELECT COUNT(DISTINCT DEPT)
FROM STUDENT
WHERE DEPT = '전산과';
```

'DEPT'의 개수를 표시하되, 같은 'DEPT' 속성의 값은 한 번만 계산한다.
<STUDENT> 테이블을 대상으로 검색한다.
'DEPT'가 "전산과"인 자료만을 대상으로 검색한다.

※ <STUDENT> 테이블에서 'DEPT' 속성의 값이 '전산과'인 튜플에 대해 중복을 제거하고 개수를 세므로 1개의 튜플이 검색됩니다.

[문제 18]

다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

데이터 마이닝은 데이터 웨어하우스에 저장된 데이터 집합에서 사용자의 요구에 따라 유용하고 가능성 있는 정보를 발견하기 위한 기법이다.

[문제 19]

(대기 시간 + 서비스 시간) / 서비스 시간

[문제 20]

50 75 85 95 100

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 콤마를 넣어 50, 75, 85, 95, 100으로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.



해설

문제의 코드는 버블 정렬 알고리즘을 이용하여 배열 `a`에 저장된 값을 오름차순으로 정렬한 후 출력하는 프로그램이다. 버블 정렬 알고리즘은 첫 번째 자료와 두 번째 자료를, 두 번째 자료와 세 번째 자료를, 세 번째와 네 번째를, ... 이런 식으로 (마지막-1)번째 자료와 마지막 자료를 비교하여 교환하면서 자료를 정렬한다. 1회전을 수행하고 나면 가장 큰 자료가 맨 뒤로 이동하므로 2회전에서는 맨 끝에 있는 자료는 정렬에서 제외되고, 2회전을 수행하고 나면 끝에서 두 번째 자료까지는 정렬에서 제외된다. 이렇게 정렬을 1회전 수행할 때마다 정렬에서 제외되는 데이터가 하나씩 늘어난다.

모든 C 프로그램은 반드시 `main()` 함수부터 시작해야 한다.

```
main() {
  ❶ int a[] = { 85, 75, 50, 100, 95 };
  ❷ align(a);
    for (int i = 0; i < 5; i++)
      printf("%d ", a[i]);
}
```

❶ 배열을 선언할 때 사용할 개수를 생략하고 초기값을 지정하면, 초기값으로 지정된 값의 수와 같은 크기의 배열이 선언된다.

	a[0]	a[1]	a[2]	a[3]	a[4]
배열 a	85	75	50	100	95

❷ `a`를 인수로 하여 함수 `align()`을 호출한다. 인수로 배열의 이름을 지정하면 배열의 시작 주소가 인수로 전달된다. 그러니까 `align(a)`는 `align(&a[0])`과 같은 의미이다.

```
  ❸ void align(int a[]) {
  ❹ int temp;
  ❺ for (int i = 0; i < 4; i++)
  ❻ for (int j = 0; j < 4 - i; j++)
  ❼ if (a[j] > a[j+1]) {
  ❽     temp = a[j];
  ❾     a[j] = a[j+1];
  ❿     a[j+1] = temp;
  ⓫ }
  ⓫ }
```

❸ ❷에서 `align(a)`라고 했으므로 정수형 배열 `a`는 `main()` 함수의 `a` 배열의 시작 주소를 받는다.

	a[0]	a[1]	a[2]	a[3]	a[4]
배열 a	85	75	50	100	95

❹ 정수형 변수 `temp`를 선언한다.

❺ 반복 변수 `i`가 0에서 시작하여 1씩 증가하면서 4보다 작은 동안 ❻번을 반복하여 수행한다.

❻ 반복 변수 `j`가 0에서 시작하여 1씩 증가하면서 4-`i`보다 작은 동안 ❼번을 반복하여 수행한다.

❼ `a[j]`가 `a[j+1]`보다 크면 ❽~❿ 사이의 문장을 실행한다.



반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

i	j	a[j]	a[j+1]	temp	배열 a
0	0	85	75	85	<div> <div>a[0] a[1] a[2] a[3] a[4]</div> <div>85 75 50 100 95</div> <div>75 85 85 95 100</div> <div>50</div> </div>
	1	75	85	85	
	2	85	50	100	
	3	50	85		
	4	85	100		
1	0	75	50	75	<div> <div>a[0] a[1] a[2] a[3] a[4]</div> <div>75 50 85 95 100</div> <div>50 75</div> </div>
	1	50	75		
	2	75	85		
	3	85	95		
2	0	50	75		<div> <div>a[0] a[1] a[2] a[3] a[4]</div> <div>50 75 85 95 100</div> </div>
	1	75	85		
	2				
3	0	50	75		<div> <div>a[0] a[1] a[2] a[3] a[4]</div> <div>50 75 85 95 100</div> </div>
	1				
4					

⑪ #문의 끝이다.

⑫ 함수를 마치고 align(a) 함수를 호출했던 main() 함수로 제어를 옮긴다.

```
main( ) {
    int a[] = { 85, 75, 50, 100, 95 };
    align(a);
    ⑬ for (int i = 0; i < 5; i++)
    ⑭ printf("%d ", a[i]);
}
```

⑬ 반복 변수 i가 0에서 시작하여 1씩 증가하면서 5보다 작은 동안 ⑭번을 반복 수행한 후 프로그램을 종료한다.

⑭ a[i]의 값을 정수로 출력한 후 한 칸을 띄운다.

결과 50 75 85 95 100