



시험에 나오는것만 공부한다!

**시나공시리즈**

2020년 실기 시험 대비  
정보처리기능사 실기 모의고사 04회



#### 저작권 안내

이 자료는 시나공 카페 회원을 대상으로 하는 자료로서 개인적인 용도로만 사용할 수 있습니다. 허락 없이 복제하거나 다른 매체에 옮겨 실을 수 없으며, 상업적 용도로 사용할 수 없습니다.

#### \*\*\* 수험자 유의사항 \*\*\*

1. 시험 문제지를 받는 즉시 응시하고자 하는 종목의 문제지가 맞는지를 확인하여야 합니다.
2. 시험 문제지 총면수·문제번호 순서·인쇄상태 등을 확인하고, 수험번호 및 성명을 답안지에 기재하여야 합니다.
3. 문제 및 답안(지), 채점기준은 일절 공개하지 않으며 자신이 작성한 답안, 문제 내용 등을 수험표 등에 이기 ( 옮겨 적는 행위 ) 등은 관련 법 등에 의거 불이익 조치 될 수 있으니 유의하시기 바랍니다.
4. 답안지 작성은 반드시 검정색 필기구만 사용하여야 합니다. ( 그 외 연필류, 유색 필기구 등을 사용한 답안은 채점하지 않으며 0점 처리됩니다. )
5. 답란(답안 기재란)에는 문제와 관련 없는 불필요한 낙서나 특이한 기록사항 등을 기재하여서는 안되며 부정의 목적으로 특이한 표식을 하였다고 판단될 경우에는 모든 문항이 0점 처리됩니다.
6. 답안을 정정할 때에는 반드시 정정부분을 두 줄(=)로 그어 표시하여야 하며, 두 줄로 굿지 않은 답안은 정정하지 않은 것으로 간주합니다. ( 수정테이프, 수정액 사용불가 )
7. 답안의 한글 또는 영문의 오타자는 오답으로 처리됩니다. 단, 답안에서 영문의 대·소문자 구분, 띄어쓰기는 여부에 관계 없이 채점합니다.
8. 계산 또는 디버깅 등 계산 연습이 필요한 경우는 <문 제> 아래의 연습란을 사용하시기 바라며, 연습란은 채점대상이 아닙니다.
9. 문제에서 요구한 가지 수(항수) 이상을 답란에 표기한 경우에는 답안기재 순으로 요구한 가지 수(항수)만 채점하고 한 항에 여러 가지를 기재하더라도 한 가지로 보며 그 중 정답과 오답이 함께 기재란에 있을 경우 오답으로 처리됩니다.
10. 한 문제에서 소문제로 파생되는 문제나, 가지수를 요구하는 문제는 대부분의 경우 부분채점을 적용합니다. 그러나 소문제로 파생되는 문제 내에서의 부분 배점은 적용하지 않습니다.
11. 답안은 문제의 마지막에 있는 답란에 작성하여야 합니다.
12. 부정 또는 불공정한 방법(시험문제 내용과 관련된 메모지사용 등)으로 시험을 치른 자는 부정행위자로 처리되어 당해 시험을 중지 또는 무효로 하고, 2년간 국가기술자격검정의 응시자격이 정지됩니다.
13. 시험위원이 시험 중 신분확인을 위하여 신분증과 수험표를 요구할 경우 반드시 제시하여야 합니다.
14. 시험 중에는 통신기기 및 전자기기(휴대용 전화기 등)를 지참하거나 사용할 수 없습니다.
15. 국가기술자격 시험문제는 일부 또는 전부가 저작권법상 보호되는 저작물이고, 저작권자는 한국산업인력공단입니다. 문제의 일부 또는 전부를 무단 복제, 배포, 출판, 전자출판 하는 등 저작권을 침해하는 일체의 행위를 금합니다.

※ 수험자 유의사항 미준수로 인한 채점상의 불이익은 수험자 본인에게 전적으로 책임이 있음

**문제 1** UNIX 시스템의 셸(Shell) 기본 명령어 중 하나로 파일 소유자와 그룹을 변경하는데 사용하는 명령어를 쓰시오. (5점)

답 :

**문제 2** 다음 설명에 해당하는 기능을 쓰시오. (5점)

다른 여러 응용 프로그램에서 작성된 문자나 그림 등의 개체(Object)를 현재 작성 중인 문서에 자유롭게 연결하거나 삽입하여 편집할 수 있게 하는 기능이다. 연결된 이미지를 원본 앱에서 수정하거나 편집하면 그 내용이 그대로 해당 문서에도 반영된다.

답 :

**문제 3** 데이터베이스에서 하나의 속성(Attribute)이 취할 수 있는 같은 타입의 원자값들의 집합으로, 실제 속성 값이 나타날 때 그 값의 허용 여부를 시스템이 검사하는데에도 사용되는 릴레이션의 구성 요소를 쓰시오. (5점)

답 :

**문제 4** 데이터베이스에 대한 다음 설명에서 괄호에 공통으로 들어갈 알맞은 용어를 쓰시오. (5점)

( )는 시스템 그 자체에 관련이 있는 다양한 객체에 관한 메타 데이터(Meta-Data)를 포함하는 시스템 데이터베이스이다. ( ) 내의 각 테이블은 사용자를 포함하여 DBMS에서 지원하는 모든 데이터 객체에 대한 정의나 명세에 관한 정보를 유지 관리하는 시스템 테이블이며, 데이터 사전(Data Dictionary)에 저장되기 때문에 좁은 의미로는 데이터 사전이라고도 한다.

답 :

---

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 5** 다음 C 언어로 구현한 프로그램에 5가 입력되었다고 가정했을 때 그 실행 결과를 쓰시오. (5점)

```
#include <stdio.h>
main() {
    int a, b, c;
    scanf("%d", &a);
    b = a << 1;
    c = b % 3 | a + 1;
    printf("%d", c);
}
```

답 :

**문제 6** 네트워크(Network)에 관한 다음 설명에 해당하는 용어를 영문 풀네임(Full-name) 또는 약어로 쓰시오. (5점)

- 국가와 국가 혹은 대륙과 대륙 등과 같이 멀리 떨어진 사이트들을 연결하여 구성한다.
- 사이트 간의 거리가 멀기 때문에 통신 속도가 느리고, 에러 발생률이 높다.
- 일정한 지역에 있는 사이트들을 근거리 통신망으로 연결한 후 각 근거리 통신망을 연결하는 방식을 사용한다.

답 :

**문제 7** 애플리케이션 테스트에 대한 다음 설명이 가리키는 용어를 쓰시오. (5점)

이미 테스트된 프로그램의 테스트를 반복하는 것으로, 통합 테스트로 인해 변경된 모듈이나 컴포넌트에 새로운 오류가 있는지 확인하는 테스트이다. 수정한 모듈이나 컴포넌트가 다른 부분에 영향을 미치는지, 오류가 생기지 않았는지 테스트하여 새로운 오류가 발생하지 않음을 보증하기 위해 반복 테스트하며, 주로 모든 애플리케이션의 기능을 수행할 수 있는 대표적인 테스트 케이스나, 변경에 의한 파급 효과가 높은 부분, 또는 실제 수정이 발생한 모듈 또는 컴포넌트에 대한 테이트 케이스가 테스트 대상으로 선정된다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 8** 다음 <회원>과 <강의> 테이블을 이용하여 <강의> 테이블의 '강의명'이 "테니스"로 시작하는 강의를 듣는 회원의 수를 검색하는 <SQL문>을 완성하시오. (5점)

<회원>

회원코드	회원명	신청강의
A01	이철중	E01
A02	김요한	E04
A03	박인중	E02
A04	최요섭	E03
A05	한나라	E01
B01	박인서	E04
B02	구선영	E02

<강의>

강의코드	강의명	강사명
E01	테니스 초급	차시애
E02	테니스 중급	곽성진
E03	서예 초급	이동예
E04	영어회화 초급	정소빈
E05	영어회화 중급	이종완

<SQL문>

```
SELECT COUNT(*) FROM 회원 WHERE 신청강의 ( ① ) (
    SELECT 강의코드 FROM 강의 WHERE 강의명 ( ② ) '테니스%');
```

답

- ①
- ②

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 9** 다음 정규화(Normalization)에 대한 설명에서 괄호(①~②)에 들어갈 알맞은 답을 쓰시오. (5점)

데이터를 데이터베이스에 저장할 때 불필요하게 중복되어 릴레이션 조작 시 예기치 못한 곤란한 현상이 발생하는 것을 ( ① )(이)라고 한다. 의도와는 상관없이 원하지 않은 값들이 삽입되거나, 삭제되거나, 갱신 시 모순이 발생하는 현상들이 모두 ( ① )에 해당한다.

이러한 ( ① )을(를) 방지하기 위해 데이터들의 중복성 및 종속성을 배제하는 정규화를 수행한다. 테이블에 속한 모든 속성의 도메인이 원자 값만으로 되어 있는 정규형을 1NF라고 한다. 아래 그림은 1NF에서 ( ② )을(를) 제거하여 2NF를 만드는 과정이다.

<Table R>

A(key)	B(key)	C	D
A345	1001	Seoul	Pmre
D347	1001	Busan	Preo
A210	1007	Gwangju	Ciqen
A345	1007	Seoul	Esto
B230	1007	Busan	Loid
D347	1201	Busan	Preo

<Table R의 함수적 종속 관계>

A, B → C, D
A → C



<Table R1>

A(key)	B(key)	D
A345	1001	Pmre
D347	1001	Preo
A210	1007	Ciqen
A345	1007	Esto
B230	1007	Loid
D347	1201	Preo

<Table R2>

A(key)	C
A345	Seoul
D347	Busan
A210	Gwangju
B230	Busan

<Table R>에서 함수적 종속관계를 보았을 때, C는 key에 해당하는 A와 B중 A에만 종속되는 ( ② ) 이므로 C를 테이블에서 분리하면 <Table R1>과 <Table R2>와 같이 2NF에 해당하는 테이블이 완성된다.

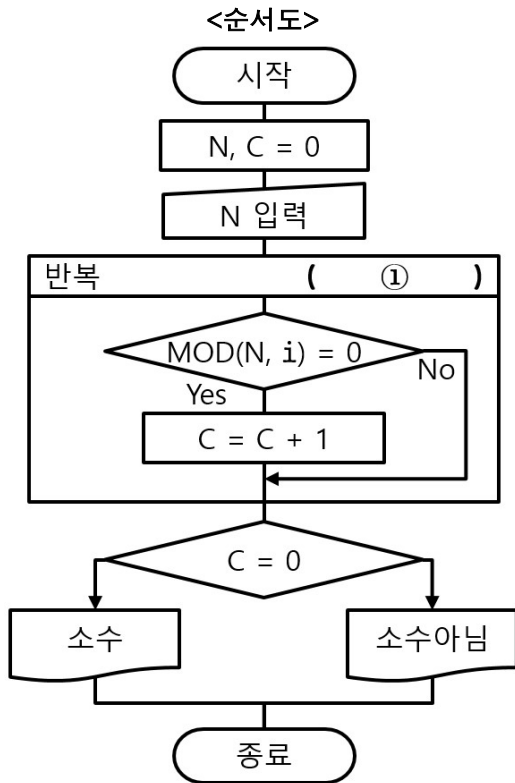
답

- ①
- ②

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

**문제 10** 다음 <순서도>와 <코드>는 입력받은 수의 소수를 판별하는 프로그램이다. 프로그램을 분석하여 <순서도>의 괄호에(①~②)에 들어갈 알맞은 답을 쓰시오. (5점)



**<코드>**

```

#include <stdio.h>

main()
{
    int n, c = 0;
    scanf("%d", &n);
    for (int i = 2; i < n; i++)
        if ( ( ② ) )
            c++;

    if (c == 0)
        printf("소수");
    else
        printf("소수아님");
}
  
```

답

- ①
- ②

**문제 11** OSI(Open System Interconnection) 전 계층에서 프로토콜(Protocol) 구조가 다른 네트워크의 연결을 수행하는 기기로, 각 계층 간 데이터 타입 변환, 주소 변환, 프로토콜 변환 등의 다양한 작업을 수행하며 네트워크들 사이에서 출입구 역할을 수행하는 네트워크 장비의 명칭을 쓰시오. (5점)

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 12** 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```
#include <stdio.h>
#define s 8
void complement(int b[]) {
    for (int i = 0; i < s; i++)
        b[i] = 1 - b[i];
}
void add(int b[]) {
    for (int i = s - 1; i >= 0; i--) {
        if (++b[i] < 2)
            break;
        b[i] -= 2;
    }
}
main() {
    int b[s] = { 0,0,0,1,1,0,1,0 };
    complement(b);
    add(b);
    for (int i = 0; i < s; i++)
        printf("%d", b[i]);
}
```

답 :

---

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 13** 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```
public class Test {  
    public static void main(String[] args) {  
        System.out.print("Ex");  
        System.out.println("change");  
        System.out.printf("%ctownc", 'A', 'B');  
    }  
}
```

답

**문제 14** <학생> 테이블에서 '이름'이 "민수"인 튜플을 삭제하는 <SQL문>을 완성하십시오. (5점)

<SQL문>

(    ①    ) (    ②    ) 학생 WHERE 이름 = '민수';

답

- ①
- ②

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.



**문제 15** 다음은 문자열 변수 s에서 “o”의 개수를 구하는 Java 프로그램이다. 프로그램을 분석하여 빈 칸에 들어갈 알맞은 코드를 쓰시오. (5점)

```
public class Test {
    public static void main(String[] args) {
        String s = "Chocolate Fondue";
        int cnt = 0;
        for (int i = 0; i < s.length(); i++)
            if (  == 'o')
                cnt++;
        System.out.print(cnt);
    }
}
```

답 :

**문제 16** 다음 <student> 테이블을 참고하여 ‘name’ 속성으로 ‘idx\_name’이라는 인덱스를 생성하는 <SQL문>을 완성하십시오. (5점)

<student>

stid	name	score	deptid
2001	brown	85	PE01
2002	white	45	EF03
2003	black	67	UW11

<SQL문>

CREATE INDEX idx\_name (  ) student(name);

답 : ON

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 17** 다음 Java 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (5점)

```
public class Test {  
    public static void main(String[] args) {  
        int i = 1, sum = 0;  
        do {  
            sum += i % 4 == 0 ? 1 : 0;  
            i++;  
        } while (i < 110);  
        System.out.printf("%d", sum);  
    }  
}
```

답 :

**문제 18** OSI 7계층의 네트워크 계층에서 속하는 프로토콜로 TCP/IP 기반의 인터넷 통신 서비스에서 인터넷 프로토콜(IP)과 조합하여 통신중에 발생하는 오류의 처리와 전송 경로 변경 등을 위한 제어 메시지를 관리하는 역할을 수행하는 프로토콜을 쓰시오. (5점)

답 :

**문제 19** 애플리케이션 테스트에 대한 다음 설명에서 괄호에 들어갈 알맞은 용어를 쓰시오. (5점)

애플리케이션 테스트에서는 잡초나 해충들이 특정 약제에 내성이 생기는 것과 같이 동일한 테스트 케이스로 동일한 테스트를 반복하면 더 이상 결함이 발견되지 않는 ( ) 현상이 발생한다. ( )를 방지하기 위해서 테스트 케이스를 지속적으로 보완 및 개선해야 한다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

**문제 20** 다음은 배열 a에 <그림>과 같이 숫자를 저장하는 Java 프로그램이다. 프로그램을 분석하여 빈 칸에 들어갈 알맞은 코드를 쓰시오. (5점)

<그림>

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

```
public class Test {  
    public static void main(String[] args) {  
        int a[][] = new int[5][5];  
        int j = 0, sw = 1, k = 1;  
        for (int i = 0; i < 5; i++) {  
            while (j <= 4 && j >= 0) {  
                a[i][j] = k++;  
                j += sw;  
            }  
              
            j += sw;  
        }  
    }  
}
```

답 :

---

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

[문제 1]

chown

[문제 2]

※ 다음 중 하나를 쓰면 됩니다.

OLE, Object Linking and Embedding

[문제 3]

※ 다음 중 하나를 쓰면 됩니다.

도메인, Domain

[문제 4]

※ 다음 중 하나를 쓰면 됩니다.

시스템 카탈로그, System Catalog

[문제 5]

7

[해설]

```
#include <stdio.h>
main() {
    ❶ int a, b, c;
    ❷ scanf("%d", &a);
    ❸ b = a << 1;
    ❹ c = b % 3 | a + 1;
    ❺ printf("%d", c);
}
```

모든 C 언어 프로그램은 반드시 main 함수에서 시작한다.

❶ 정수형 변수 a, b, c를 선언한다.

❷ 정수를 입력받아 a에 저장한다. 문제에서 5가 입력되었다고 하였으므로 5가 저장된다.

❸ <<는 왼쪽 시프트 연산자이므로, a에 저장된 값 5를 왼쪽으로 1비트 이동시킨 다음 그 값을 b에 저장한다. int는 4Byte이므로 4Byte 2진수로 변환하여 계산하면 된다.

• 4바이트에 5를 2진수로 표현하면 다음과 같다.

	32	31	30	29	...	9	8	7	6	5	4	3	2	1
5	0	0	0	0	...	0	0	0	0	0	0	1	0	1
					...		$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
					...		128	64	32	16	8	4	2	1

부호 비트

• 부호를 제외한 전체 비트를 왼쪽으로 1비트 이동시킨다. 양수이므로 패딩 비트(빈자리)에는 0이 채워진다.

	32	31	30	29	...	9	8	7	6	5	4	3	2	1	
10	0	0	0	0	...	0	0	0	0	0	1	0	1	0	패딩 비트
	부호 비트				...		$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	
					...		128	64	32	16	8	4	2	1	

• b에 저장된 값을 10진수로 변환하면 10이다.

④  $c = b \% 3 \mid a + 1;$

①                      ②  
\_\_\_\_\_

③

① b의 값이 10이므로,  $10 \% 3 = 1$

② a의 값이 5이므로,  $5 + 1 = 6$

③ c에 ①과 ②를 |(비트 or) 연산한 값을 저장한다. |(비트 or)는 두 비트 중 한 비트라도 1이면 1이 되는 비트 연산자이다.

①(1) = 0001

②(6) = 0110

| = 0111 = 7

⑤ c의 값을 출력한다.

결과 **7**

[문제 6]

※ 다음 중 하나를 쓰면 됩니다.

WAN, Wide Area Network

[문제 7]

※ 다음 중 하나를 쓰면 됩니다.

회귀 테스트, Regression Test

[문제 8]

① IN                      ② LIKE

[풀이]

② SELECT COUNT(*)	튜플의 개수를 표시한다.
FROM 회원	<회원> 테이블에서 검색한다.
WHERE 신청강의 IN (	'신청강의'가 IN 다음에 쓰인 하위 질의의 검색 결과에 속한 자료만을 대상으로 한다.
① SELECT 강의코드	'강의코드'를 표시한다.
FROM 강의	<강의> 테이블에서 검색한다.
WHERE 강의명 LIKE '테니스%');	'강의명'이 "테니스"로 시작하는 자료만을 대상으로 한다.

하위 질의가 있는 경우에는 하위 질의가 먼저 수행된다.

① SELECT 강의코드 FROM 강의 WHERE 강의명 LIKE '테니스%' : <강의> 테이블의 '강의코드'를 검색하되 '강의명'이 "테니스"로 시작하는 튜플만을 검색한다.

<결과>

강의코드
E01
E02

② SELECT COUNT(\*) FROM 회원 WHERE 신청강의 IN (①) : <회원> 테이블에서 튜플의 개수를 검색하되 '신청강의'가 ①의 결과에 해당하는 튜플만을 대상으로 한다. ①의 '강의코드'인 "E01"과 "E02"에 해당하는 튜플에는 "이철중", "박인중", "한나라", "구선영"이 있으므로 그 개수인 4가 출력된다.

<결과>

COUNT(*)
4

[문제 9]

※ 각 문항별로 제시된 답안 중 한 가지만 쓰면 됩니다.

- ① 이상, Anomaly
- ② 부분 함수 종속, 부분 함수적 종속, 부분적 함수 종속

[문제 10]

- ①  $i = 2, N - 1, 1$
- ②  $n \% i == 0$

[답안 작성 시 주의 사항]

C 언어에서 사용하는 변수, 함수 등의 이름은 대소문자를 구분하기 때문에 작성할 때 대소문자를 구분해서 정확히 작성해야 합니다.

[해설]

<코드>

```
#include <stdio.h>
main()
{
    ①    int n, c = 0;
    ②    scanf("%d", &n);
    ③    for (int i = 2; i < n; i++)
    ④        if (n % i == 0)
    ⑤            c++;
    ⑥    if (c == 0)
    ⑦        printf("소수");
        else
    ⑧        printf("소수아님");
}
```

모든 C 언어 프로그램은 반드시 main 함수에서 시작한다.

- ❶ 정수형 변수 n, c를 선언하고, c를 0으로 초기화한다.
- ❷ 정수를 입력받아 n에 저장한다.
- ❸ 반복 변수 i가 2부터 1씩 증가하면서 n보다 작은 동안 ❹~❺번을 반복 수행한다.
- ❹ n을 i로 나눈 나머지가 0이면 ❺번을 수행하고, 아니면 반복문의 처음인 ❸번으로 이동한다.
- ❺ 'c = c + 1;'과 동일하다. c에 1씩 누적시킨다.
- ❻ c가 0이면 ❼번을 수행하고, 아니면 ❽번을 수행한다.
- ❼ 문자열 소수 를 출력한다.
- ❽ 문자열 소수아님 을 출력한다.

n에 5가 입력되었다고 가정한 프로그램에서 변수들의 값의 변화는 다음과 같다.

n	i	n % i	c	출력
5			0	소수
	2	1		
	3	2		
	4	1		
	5			

#### <순서도>

- 순서도와 코드를 비교했을 때 ❶번에 들어가야 하는 부분의 코드는 다음과 같습니다.

```
for (int i = 2; i < n; i++)
```

- n을 5라고 가정했을 때 for에 속한 코드는 n이 2, 3, 4일 때 총 3번이 수행됩니다.
- 만약 순서도에도 for문과 동일하게 **i = 2, N, 1**이라고 쓰게되는 경우 순서도는 [반복] 도형에 속한 처리문들을 i가 2, 3, 4, 5일 때 총 4번을 수행하게 되므로 최대값 n에서 1을 감소시켜 **i = 2, N-1, 1**로 작성해야 합니다.

#### [문제 11]

※ 다음 중 하나를 쓰면 됩니다.

게이트웨이, Gateway

#### [문제 12]

11100110

#### [답안 작성 시 주의 사항]

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어, 출력값들을 십자로 구분하여 1,1,1,0,0,1,1,0과 같이 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

```
#include <stdio.h>
#define s 8          s를 상수 8로 정의하는 명령문이다.
                    프로그램 전체에서 s는 상수 8과 동일하게 사용된다.
❸ void complement(int b[]) {
❹     for (int i = 0; i < s; i++)
❺         b[i] = 1 - b[i];
}
❷ void add(int b[]) {
❸     for (int i = s - 1; i >= 0; i--) {
❹         if (++b[i] < 2)
❺             break;
❻         b[i] -= 2;
}
}
main() {
❶     int b[s] = { 0,0,0,1,1,0,1,0 };
❷     complement(b);
❸     add(b);
❹     for (int i = 0; i < s; i++)
❺         printf("%d", b[i]);
}
```

모든 C 언어 프로그램은 반드시 main 함수에서 시작한다.

❶ 8개의 요소를 갖는 배열 b를 선언하고 초기화 한다.

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
b	0	0	0	1	1	0	1	0

❷ 배열 b의 주소를 인수로 complement 함수를 호출한다.

※ 배열의 값이 아닌 주소가 인수로 전달되었으므로 다른 함수에서의 배열 값의 변화는 main 함수의 배열에도 영향을 준다.

❸ complement 함수의 시작점이다. ❷번에서 전달받은 값을 배열 b가 받는다.

❹ 반복 변수 i가 0부터 1씩 증가하면서 s보다 작은 동안 ❺번을 반복 수행한다.

❺ 1에서 b[i]를 뺀 값을 b[i]에 저장한다. 반복이 종료되면 함수를 종료하고 함수를 호출했던 ❷번의 다음 줄인 ❻번으로 이동한다.

반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

i	b[]							
0	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
1	0	0	0	1	1	0	1	0
2	1	1	1	0	0	1	0	1
3								
4								
5								
6								
7								
8								



- ⑥ 배열 b의 주소를 인수로 add 함수를 호출한다.
- ⑦ add 함수의 시작점이다. ⑥번에서 전달받은 값을 배열 b가 받는다.
- ⑧ 반복 변수 i가 s-1(7)부터 1씩 감소하면서 0보다 크거나 같은 동안 ⑨~⑪번을 반복 수행한다.
- ⑨ b[i]에 전치증가 연산자가 있으므로 b[i]의 값에 1을 먼저 더한 후, b[i]가 2보다 작으면 ⑩번을 수행하고 아니면 ⑪번으로 이동한다.
- ⑩ for문을 벗어난다. 함수가 종료되었으므로, 함수를 호출했던 ⑥번의 다음 줄인 ⑫번으로 이동한다.
- ⑪ 'b[i] = b[i] - 2;'와 동일하다. b[i]에서 2를 뺀 값을 b[i]에 저장한다.
- 반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

i	b[]							
7	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
6	1	1	1	0	0	1	0	1
							1	0

- ⑫ 반복 변수 i가 0부터 1씩 증가하면서 s보다 작은 동안 ⑬번을 반복 수행한다.
- ⑬ b[i]의 값을 출력한다.

결과 **11100110**

### [문제 13]

Exchange

Ato

B

### [답안 작성 시 주의 사항]

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어, 출력값들을 공백으로 구분하여 Ex change Ato B와 같이 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

### [해설]

```
public class Test {
    public static void main(String[] args) {
        ①      System.out.print("Ex");
        ②      System.out.println("change");
        ③      System.out.printf("%ctowN%c", 'A', 'B');
    }
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

- ① 문자열 Ex를 출력한다.

결과 **Ex**

- ② 문자열 change를 출력하고 커서를 다음 줄의 처음으로 옮긴다.

결과 **Exchange**

- ③ 문자 A와 문자 to를 출력한 후 커서를 다음 줄의 처음으로 옮긴다. 이어서 문자 B를 출력한다.

결과 **Exchange**

**Ato**

**B**

[문제 14]

- ① DELETE                      ② FROM

[풀이]

DELETE	삭제하라
FROM 학생	<학생> 테이블을 대상으로 하라.
WHERE 이름 = '민수';	'이름'이 "민수"인 자료만을 대상으로 한다.

[문제 15]

s.charAt(i)

[답안 작성 시 주의 사항]

Java에서 사용하는 변수, 메소드 등의 이름은 대소문자를 구분하기 때문에 작성할 때 대소문자를 구분해서 정확하게 작성해야 합니다.

[해설]

```
public class Test {
    public static void main(String[] args) {
        ❶      String s = "Chocolate Fondue";
        ❷      int cnt = 0;
        ❸      for (int i = 0; i < s.length(); i++)
        ❹          if (s.charAt(i) == 'o')
        ❺              cnt++;
        ❻      System.out.print(cnt);
    }
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

❶ 문자열 변수 s를 선언하고 Chocolate Fondue로 초기화한다.

❷ 정수형 변수 cnt를 선언하고 0으로 초기화한다.

❸ 반복 변수 i가 0부터 1씩 증가하면서 s의 길이 16보다 작은 동안 ❹~❺번을 반복 수행한다.

※ length() : 문자열의 길이를 반환하는 메소드

❹ s에서 i번째에 있는 문자가 o이면 ❺번을 수행하고 아니면 반복문의 처음인 ❸번으로 이동한다.

※ charAt() : 문자열에서 지정된 위치의 문자를 반환하는 메소드

❺ 'cnt = cnt + 1;'과 동일하다. cnt를 1씩 누적시킨다.

반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]
s	C	h	o	c	o	l	a	t	e		F	o	n	d	u	e

i	s.charAt(i)	cnt
		0
0	C	
1	h	
2	o	1
3	c	
4	o	2
5	l	
6	a	
7	t	
8	e	
9		
10	F	
11	o	3
12	n	
13	d	
14	u	
15	e	
16		

⑥ cnt의 값을 출력한다.

결과 **3**

[문제 16]

ON

[풀이]

```
CREATE INDEX idx_name
ON student(name);
```

'idx\_name'이라는 이름의 인덱스를 생성한다.  
<student> 테이블의 'name' 속성을 사용한다.

[문제 17]

27

[해설]

```
public class Test {
    public static void main(String[] args) {
①        int i = 1, sum = 0;
②        do {
③            sum += i % 4 == 0 ? 1 : 0;
④            i++;
⑤        } while (i < 110);
⑥        System.out.printf("%d", sum);
    }
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

❶ 정수형 변수 i와 sum을 선언하고, 각각 1과 0으로 초기화한다.

❷ do~while 반복문의 시작점이다. ❸~❹번 문장을 반복하여 수행한다.

❸ 'sum = sum + i % 4 == 0 ? 1 : 0;'과 동일하다. i를 4로 나눈 나머지가 0이면 sum에 1을 더하고, 아니면 0을 더한다.

❹ 'i = i + 1;'과 동일하다. i에 1씩 누적시킨다.

❺ i가 110보다 작은 동안 ❸~❹번 문장을 반복하여 수행한다.

반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

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

❻ sum의 값을 출력한다.

결과 27

#### [문제 18]

※ 다음 중 하나를 쓰면 됩니다.

ICMP, Internet Control Message Protocol

#### [문제 19]

※ 다음 중 하나를 쓰면 됩니다.

살충제 패러독스, Pesticide Paradox

[문제 20]

sw \*= -1;

※ 답안 작성 시 주의 사항

sw 변수가 for문이 반복될 때마다 1, -1, 1, -1, 1로 변화할 수 있는 어떠한 식도 정답이 될 수 있습니다. 예를 들어 **sw = sw \* -1**로 작성하거나, 음수 기호만 붙여 **sw = -sw**로 작성해도 답이 됩니다.

[해설]

```
public class Test {
    public static void main(String[] args) {
        ❶      int a[][] = new int[5][5];
        ❷      int j = 0, sw = 1, k = 1;
        ❸      for (int i = 0; i < 5; i++) {
        ❹          while (j <= 4 && j >= 0) {
        ❺              a[i][j] = k++;
        ❻              j += sw;
        ❼          }
        ❼          sw *= -1;
        ❽          j += sw;
        ❹      }
    }
}
```

- ❶ 5행 5열의 크기를 갖는 정수형 배열 a를 선언한다.
- ❷ 정수형 변수 j, sw, k를 선언하고 각각 0, 1, 1로 초기화한다.
- ❸ 반복 변수 i가 0부터 1씩 증가하면서 5보다 작은 동안 ❹~❽번을 반복 수행한다.
- ❹ j가 4보다 작거나 같고 0보다 크거나 같은 동안 ❺~❻번을 반복 수행한다.
- ❺ a[i][j]에 k의 값을 저장한다. k는 후치 증가 연산이므로 연산을 마치면 1이 증가한다.
- ❻ 'j = j + sw;'와 같다. j에 sw의 값을 누적한다.
- ❼ 'sw = sw \* -1;'과 같다. sw의 값에 -1을 곱한 값을 sw에 저장한다.
- ❽ 'j = j + sw;'와 같다. j에 sw의 값을 누적한다.

반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

i	j	sw	k	배열 a																														
	0	1	1																															
0	1	-1	2	<table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	3	4	5																									
	1		2		3	4	5																											
2	3																																	
3	4																																	
4	5																																	
5	6																																	
4																																		
1	3	1	7	<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></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table>	1	2	3	4	5	10	9	8	7	6																				
	1		2		3	4	5																											
	10		9		8	7	6																											
2	8																																	
1	9																																	
0	10																																	
-1	11																																	
0																																		

2	1 2 3 4 5 4	-1	12 13 14 15 16	<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></td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td><td></td></tr> </table>	1	2	3	4	5	10	9	8	7	6	11	12	13	14	15										
1	2	3	4	5																									
10	9	8	7	6																									
11	12	13	14	15																									
3	3 2 1 0 -1 0	1	17 18 19 20 21	<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></td><td></td><td></td><td></td><td></td></tr> </table>	1	2	3	4	5	10	9	8	7	6	11	12	13	14	15	20	19	18	17	16					
1	2	3	4	5																									
10	9	8	7	6																									
11	12	13	14	15																									
20	19	18	17	16																									
4	1 2 3 4 5 4	-1	22 23 24 25 26	<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																									
5																													