



시험에 나오는것만 공부한다!

시나공시리즈

기출문제

2023년 3회 정보처리기사 실기



정보처리기사 실기 시험은 한국산업인력공단에서 문제를 공개하지 않아 문제 복원에 많은 어려움이 있습니다. 다음에 제시된 문제는 시험을 치른 학생들의 기억을 토대로 복원한 것이므로, 일부 내용이나 문제별 배점이 실제 시험과 다를 수 있음을 알립니다.

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. 수험자 인적사항 및 답안작성(계산식 포함)은 흑색 기구만 사용하되, 동일한 한 가지 색의 필기구만 사용하여야 하며 흑색을 제외한 유색 필기구 또는 연필류를 사용하거나 2가지 이상의 색을 혼합 사용하였을 경우 그 문항은 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.
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 다음 JAVA로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
class SuperObject {
    public void draw() {
        System.out.println("A");
        draw();
    }
    public void paint() {
        System.out.print('B');
        draw();
    }
}
class SubObject extends SuperObject {
    public void paint() {
        super.paint();
        System.out.print('C');
        draw();
    }
    public void draw() {
        System.out.print('D');
    }
}
public class Test {
    public static void main(String[] args) {
        SuperObject a = new SubObject();
        a.paint();
        a.draw();
    }
}
```

답 :

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

문제 2 다음 설명에 해당하는 용어를 <보기>에서 찾아 쓰시오. (5점)

- 인터넷 애플리케이션에서 사용자 인증에 사용되는 표준 인증 방법으로, 공개 API(OpenAPI)로 구현었다.
- 인터넷 사용자들끼리 웹사이트나 애플리케이션에 비밀번호를 제공하지 않고 자신에게 접근 권한을 부여하여 사용할 수 있다.
- 2010년 ETF에서 1.0이 공식 표준안으로 발표되었다.

<보기>

- | | | | |
|----------|--------|---------|--------|
| • OpenID | • IDEA | • OAuth | • SSPI |
| • SASL | • PEAP | • OIDC | • JAAS |

답 :

문제 3 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하십시오.) (5점)

```
#include <stdio.h>
main() {
    char* p = "KOREA";
    printf("1. %s\n", p);
    printf("2. %s\n", p + 1);
    printf("3. %c\n", *p);
    printf("4. %c\n", *(p + 3));
    printf("5. %c\n", *p + 4);
}
```

답

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

문제 4 다음 C 언어 프로그램과 그 <실행결과>를 분석하여 괄호에 공통으로 들어갈 알맞은 답을 쓰시오.
(5점)

```
#include <stdio.h>
struct insa {
    char name[10];
    int age;
    struct insa* impl_a;
    struct insa* impl_b;
};

main() {
    struct insa p1 = { "Kim", 28, NULL, NULL };
    struct insa p2 = { "Lee", 36, NULL, NULL };
    struct insa p3 = { "Park", 41, NULL, NULL };
    p1.impl_a = &p2;
    p2.impl_b = &p3;
    printf("%s\n", p1.impl_a(  )name);
    printf("%d", p2.impl_b(  )age);
}
```

<실행결과>

```
Lee
41
```

답 :

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

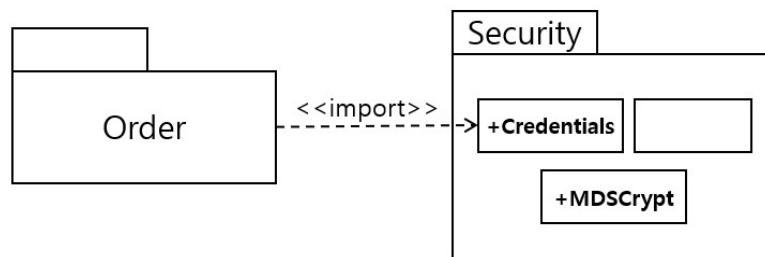
문제 5 리눅스 또는 유닉스에서 'a.txt' 파일에 대해 다음 <처리 조건>과 같이 권한을 부여하고자 한다. <처리 조건>을 준수하여 식을 완성하시오. (5점)

<처리 조건>

- 사용자에게 읽기, 쓰기, 실행 권한을 부여한다.
- 그룹에게 읽기, 실행 권한을 부여한다.
- 기타 사용자에게 실행 권한을 부여한다.
- 한 줄로 작성하고, 8진법 숫자를 이용한 명령문을 이용한다.

답 : () () a.txt

문제 6 UML 다이어그램에 대한 다음 설명에서 괄호에 공통으로 들어갈 알맞은 용어를 쓰시오. (5점)



() 다이어그램

()은(는) UML 정적 모델링의 하나로, 관련있는 객체들을 하나로 묶어 상위 개념으로 추상화한 것이다. 위의 그림과 같이 유스케이스나 클래스 등의 요소들을 그룹화하여 의존 관계를 표현하며 대규모 시스템에서 주요 요소 간의 종속성을 파악하는 데 사용한다. 시스템의 구조를 간략하게 표현할 수 있고 의존 관계를 명확하게 파악할 수 있어, 불필요한 의존 관계를 제거하거나 간략화함으로써 시스템의 복잡도를 낮추는데에도 사용할 수 있다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 7 테스트 기법 중 다음과 같이 ‘평가 점수표’를 미리 정해 놓은 후 각 영역에 해당하는 입력값을 넣고, 예상되는 출력값이 나오는지 실제 값과 비교하는 명세 기반 테스트 기법을 <보기>에서 찾아 쓰시오. (5점)

<평가 점수표>

평가점수	성적등급
90~100	A
80~89	B
70~79	C
0~69	D

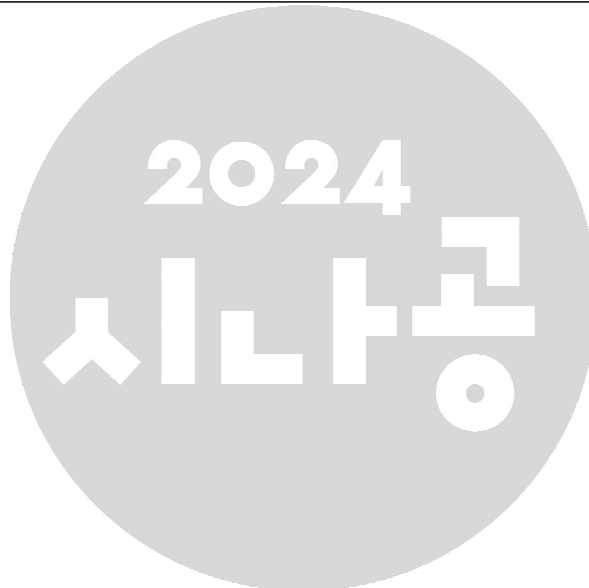
<케이스>

테스트 케이스	1	2	3	4
점수범위	0~69	70~79	80~89	90~100
입력값	60	75	82	96
예상 결과값	D	C	B	A
실제 결과값	D	C	B	A

<보기>

• Equivalence Partition	• Boundary Value Analysis	• Equivalence Value
• Cause-Effect Graph	• Error Guess	• Comparison Test
• Base Path Test	• Loop Test	• Data Flow Test

답 :



연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 8 <R>과 <S> 테이블에 대해 <SQL문>을 실행하였을 때 나타나는 결과를 작성하시오. (SQL을 실행하였을 때 출력되는 속성명과 값들을 모두 답안에 적으시오.) (5점)

<R>

A	B
1	a
2	b
3	c

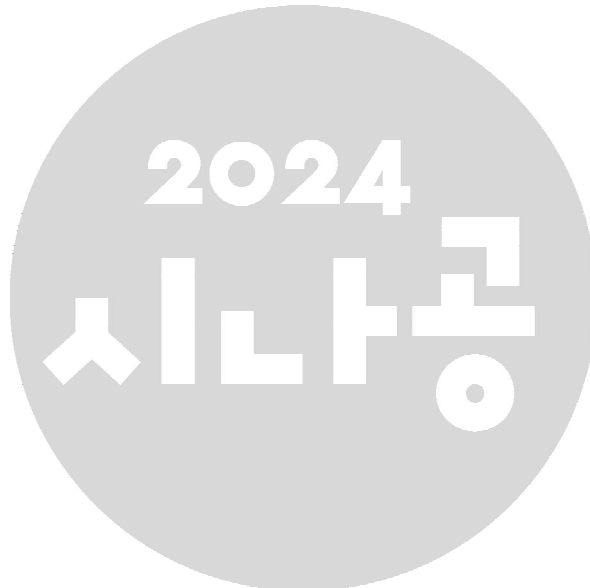
<S>

A	B
1	a
2	c
4	d

<SQL문>

```
SELECT A FROM R
UNION
SELECT A FROM S
ORDER BY A DESC;
```

답 :



연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 9 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
int isPerfectNum(int num) {
    int sum = 0;
    for (int i = 1; i < num; i++) {
        if (num % i == 0)
            sum += i;
    }
    if (num == sum) return 1;
    else return 0;
}
main() {
    int r = 0;
    for (int i = 1; i <= 100; i++)
        if (isPerfectNum(i))
            r += i;

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

답 :

문제 10 네트워크에 대한 다음 설명에 해당하는 용어를 쓰시오. (5점)

- 우리말로 번역하면 ‘네트워크 주소 변환’이라는 의미의 영문 3글자 약자이다.
- 1개의 정식 IP 주소에 다량의 가상 사설 IP 주소를 할당 및 연결하는 방식이다.
- 1개의 IP 주소를 사용해서 외부에 접속할 수 있는 노드는 어느 시점에서 1개만으로 제한되는 문제가 있으나, 이 때에는 IP 마스커레이드(Masquerade)를 이용하면 된다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 11 다음 설명에 해당하는 프로토콜을 쓰시오. (5점)

자료를 일정한 크기로 정하여 순서대로 전송하는 자료의 전송방식으로, 셀이라 부르는 53Byte의 고정 길이 패킷을 이용하여 처리가 단순하고 고속망에 적합하다. 또한 연속적으로 셀을 보낼 때 다중화를 하지 않고 셀 단위로 동기가 이루어지지만 경우에 따라 동기식 시간 분할 다중화를 사용하기도 한다. CBR, VBR의 처리가 가능하며, B-ISDN과 결합하여 서비스를 제공하기도 한다.

답 :

문제 12 다음은 오류가 발생하는 JAVA 프로그램이다. 프로그램을 분석하여 오류가 발생하는 라인을 쓰시오. (5점)

라인	코드
1	class Person {
2	private String name;
3	public Person(String val) {
4	name = val;
5	}
6	public static String get() {
7	return name;
8	}
9	public void print() {
10	System.out.println(name);
11	}
12	}
13	public class Test {
14	public static void main(String[] args) {
15	Person obj = new Person("Kim");
16	obj.print();
17	}
18	}

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 13 접근 통제(Access Control)에 대한 다음 설명에서 괄호(①~③)에 들어갈 알맞은 용어를 <보기>에서 찾아 쓰시오. (5점)

(①)	<ul style="list-style-type: none"> • 주체와 객체의 등급을 비교하여 접근 권한을 부여하는 방식이다. • 시스템이 접근통제 권한을 지정한다. • 데이터베이스 객체별로 보안 등급을 부여할 수 있다. • 사용자별로 인가 등급을 부여할 수 있다.
(②)	<ul style="list-style-type: none"> • 사용자의 역할에 따라 접근 권한을 부여하는 방식이다. • 중앙관리자가 접근통제 권한을 지정한다. • 임의 접근통제와 강제 접근통제의 단점을 보완하였다. • 다중 프로그래밍 환경에 최적화된 방식이다.
(③)	<ul style="list-style-type: none"> • 데이터에 접근하는 사용자의 신원에 따라 접근 권한을 부여하는 방식이다. • 데이터 소유자가 접근통제 권한을 지정하고 제어한다. • 객체를 생성한 사용자가 생성된 객체에 대한 모든 권한을 부여받고, 부여된 권한을 다른 사용자에게 허가할 수도 있다.

<보기>

- DAC(Discretionary Access Control)
- MAC(Mandatory Access Control)
- RBAC(Role Based Access Control)

답

- ①
- ②
- ③

2024
시나공

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 14 다음 JAVA로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
class P {
    public int calc(int n) {
        if (n <= 1) return n;
        return calc(n - 1) + calc(n - 2);
    }
}
class C extends P {
    public int calc(int n) {
        if (n <= 1) return n;
        return calc(n - 1) + calc(n - 3);
    }
}
public class Test {
    public static void main(String[] args) {
        P obj = new C();
        System.out.print(obj.calc(7));
    }
}
```

답 :

문제 15 다음 C 언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
int f(int n) {
    if (n <= 1) return 1;
    else return n * f(n - 1);
}
main() {
    printf("%d", f(7));
}
```

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

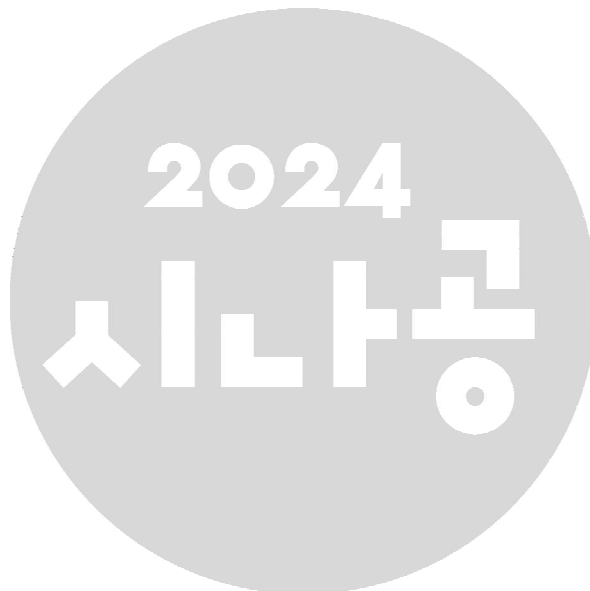
문제 16 다음 Python 프로그램과 그 <실행결과>를 분석하여 괄호에 들어갈 알맞은 예약어를 쓰시오. (<실행결과> 첫 번째 라인의 '5 10'은 입력받은 값에 해당한다.) (5점)

```
x, y = input("x, y의 값을 공백으로 구분하여 입력 : ").(      )(' ')\nprint("x의 값 :", x)\nprint("y의 값 :", y)
```

<실행결과>

```
x, y의 값을 공백으로 구분하여 입력 : 5 10\nx의 값 : 5\ny의 값 : 10
```

답 :



연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 17 클라우드에 대한 다음 설명에서 괄호(①~③)에 들어갈 알맞은 용어를 내용 중에서 찾아 쓰시오.
(5점)

클라우드 컴퓨팅은 각종 컴퓨팅 자원을 중앙 컴퓨터에 두고 인터넷 기능을 갖는 단말기로 언제 어디서나 인터넷을 통해 컴퓨터 작업을 수행할 수 있는 환경을 의미한다. 중앙 컴퓨터는 복수의 데이터 센터를 가상화 기술로 통합한 대형 데이터 센터로, 각종 소프트웨어, 데이터, 보안 솔루션 기능 등 컴퓨팅 자원을 보유하고 있다. 사용자는 키보드와 모니터, 마우스를 갖추고 통신 포트만 연결하면 업무 수행이 가능하다.

즉 클라우드 컴퓨팅은 인터넷으로 가상화된 IT 리소스를 서비스로 제공하는 것을 의미하며, 클라우드 컴퓨팅에서 가상화하여 서비스로 제공하는 대상에 따라 IaaS, PaaS, SaaS로 구분되어 진다.



답

- ①
- ②
- ③

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 18 다음 설명에 해당하는 프로토콜을 쓰시오. (5점)

- 거리 벡터 라우팅 프로토콜이라고도 불리며, 최단 경로 탐색에 Bellman-Ford 알고리즘이 사용된다.
- 소규모 동종의 네트워크 내에서는 효율적이나, 최대 홉(Hop) 수가 제한되므로 대규모 네트워크에서는 사용할 수 없다.
- 일정 시간 동안 라우팅 정보가 갱신되지 않으면 해당 경로를 이상 상태로 간주한다.

답 :

문제 19 관계 연산자에 대한 다음 설명에서 각 번호(①~④)의 연산자를 기입하는 데 사용하는 기호를 <보기>에서 찾아 쓰시오. (5점)

연산자	특징
① Join	<ul style="list-style-type: none"> • 공통 속성을 중심으로 두 개의 릴레이션을 하나로 합쳐서 새로운 릴레이션을 만드는 연산이다. • Join의 결과는 Cartesian Product(교차곱)를 수행한 다음 Select를 수행한 것과 같다.
② Project	<ul style="list-style-type: none"> • 주어진 릴레이션에서 속성 리스트(Attribute List)에 제시된 속성 값만을 추출하여 새로운 릴레이션을 만드는 연산이다. • 연산 결과에 중복이 발생하면 중복이 제거된다. • 릴레이션의 열에 해당하는 속성을 추출하는 것이므로 수직 연산자라고도 한다.
③ Select	<ul style="list-style-type: none"> • 릴레이션에 존재하는 튜플 중에서 선택 조건을 만족하는 튜플의 부분집합을 구하여 새로운 릴레이션을 만드는 연산이다. • 릴레이션의 행에 해당하는 튜플(Tuple)을 구하는 것이므로 수평 연산자라고도 한다.
④ Division	<ul style="list-style-type: none"> • $X \supset Y$인 두 개의 릴레이션 $R(X)$와 $S(Y)$가 있을 때, R의 속성이 S의 속성값을 모두 가진 튜플에서 S가 가진 속성을 제외한 속성만을 구하는 연산이다.

<보기>

• σ	• \bowtie	• \div	• \pm	• ∞	• π
------------	-------------	----------	---------	------------	---------

답

- ① Join :
- ② Project :
- ③ Select :
- ④ Division :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 20 무결성에 대한 다음 설명에서 괄호에 들어갈 알맞은 답을 쓰시오. (5점)

무결성이란 데이터베이스에 저장된 데이터 값과 그것이 표현하는 현실 세계의 실제 값이 일치하는 정확성을 의미한다. 무결성 제약 조건은 데이터베이스에 들어 있는 데이터의 정확성을 보장하기 위해 부정확한 자료가 데이터베이스 내에 저장되는 것을 방지하기 위한 제약 조건을 말한다.

<회원>

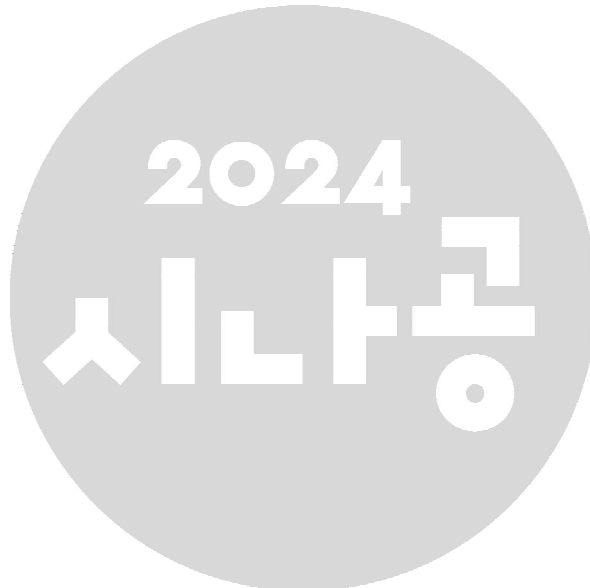
이름	주민번호	주소
kim	800212-2*****	서울
choi	820911-1*****	경기
kang	750815-1*****	인천

<결제>

결제번호	주문 상품	회원
1	rx-123	choi
2	dp-01	park
3	qiv-433	kang

위의 두 테이블에서는 <결제> 테이블의 '회원' 속성을 <회원> 테이블의 '이름' 속성을 참고하는 외래키이므로 () 무결성 제약 조건이 준수되어야 한다.

답 :



연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

[문제 1]

BDCDD

[해설]

```
class SuperObject {
    public void draw() {
        System.out.println("A");
        draw();
    }
    ⑤ public void paint() {
    ⑥         System.out.print('B');
    ⑦         draw();
        } ⑩
}
class SubObject extends SuperObject {
    ③ public void paint() {
    ④         super.paint();
    ⑪         System.out.print('C');
    ⑫         draw();
        } ⑮
    ⑧⑬⑰ public void draw() {
    ⑨⑭⑱         System.out.print('D');
        }
}
public class Test {
    public static void main(String[] args) {
    ①         SuperObject a = new SubObject();
    ②         a.paint();
    ⑯         a.draw();
        } ⑲
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

① SubObject 클래스의 생성자를 이용하여 SuperObject 클래스의 객체 변수 a를 생성한다.

② a의 paint() 메소드를 호출한다. ③번으로 이동한다.

a.paint()는 a 객체의 자료형이 SuperObject이므로 SuperObject.paint()라고 생각할 수 있지만 ①번에서 클래스 형 변환이 발생하였고, paint() 메소드가 자식 클래스에서 재정의되었으므로 SubObject 클래스의 paint() 메소드가 호출된다.

③ 반환값이 없는 paint() 메소드의 시작점이다.

④ 부모 클래스를 호출하는 예약어 super를 사용했으므로 부모 클래스의 paint() 메소드를 호출한다. ⑤번으로 이동한다.

⑤ 반환값이 없는 paint() 메소드의 시작점이다.

⑥ 화면에 B를 출력한다.

결과 B

⑦ draw() 메소드를 호출한다. draw() 메소드는 ②번의 경우와 마찬가지로 자식 클래스에서 재정의되었으므로 SubObject 클래스의 draw() 메소드가 호출된다.

⑧ 반환값이 없는 draw() 메소드의 시작점이다.

⑨ 화면에 D를 출력한다. draw() 메소드가 종료되었으므로 메소드를 호출했던 ⑦번의 다음 줄인 ⑩번으로 이

동한다.

결과 **BD**

⑩ paint() 메소드가 종료되었으므로 메소드를 호출했던 ④번의 다음 줄인 ⑪번으로 이동한다.

⑪ 화면에 C를 출력한다.

결과 **BDC**

⑫ draw() 메소드를 호출한다.

⑬ 반환값이 없는 draw() 메소드의 시작점이다.

⑭ 화면에 D를 출력한다. draw() 메소드가 종료되었으므로 메소드를 호출했던 ⑫번의 다음 줄인 ⑮번으로 이동한다.

결과 **BDCD**

⑮ paint() 메소드가 종료되었으므로 메소드를 호출했던 ②번의 다음 줄인 ⑯번으로 이동한다.

⑯ draw() 메소드를 호출한다.

⑰ 반환값이 없는 draw() 메소드의 시작점이다.

⑱ 화면에 D를 출력한다. draw() 메소드가 종료되었으므로 메소드를 호출했던 ⑯번의 다음 줄인 ⑲번으로 이동한다.

결과 **BDCDD**

⑲ main() 메소드가 종료되었으므로 프로그램이 종료된다.

[문제 2]

OAuth

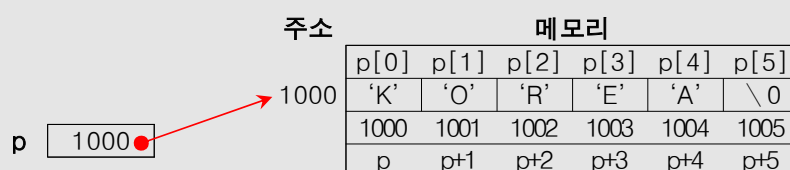
[문제 3]

1. KOREA
2. OREA
3. K
4. E
5. O

[해설]

```
#include <stdio.h>
main() {
  ① char* p = "KOREA";
  ② printf("1. %s\n", p);
  ③ printf("2. %s\n", p + 1);
  ④ printf("3. %c\n", *p);
  ⑤ printf("4. %c\n", *(p + 3));
  ⑥ printf("5. %c\n", *p + 4);
}
```

① 문자형 포인터 변수 p에 "KOREA"를 저장한다. (다음 그림에서 지정한 주소는 임의로 정한 것이며, 이해를 돕기 위해 10진수로 표현했다.)



② 1. 을 출력하고 p의 위치부터 널 문자(\0)까지의 모든 문자를 출력한다.

결과 **1. KOREA**

③ 2. 을 출력하고 p+1의 위치부터 널 문자(\0)까지의 모든 문자를 출력한다.

결과 1. KOREA
2. OREA

④ 3. 을 출력하고 p가 가리키는 곳의 문자 K를 출력한다.

결과 1. KOREA
2. OREA
3. K

⑤ 4. 을 출력하고 p+3이 가리키는 곳의 문자 E를 출력한다.

결과 1. KOREA
2. OREA
3. K
4. E

⑥ 5. 을 출력하고 p가 가리키는 곳의 값에 4를 더한 후 출력한다. p가 가리키는 곳의 값은 문자 'K'인데, 알파벳 문자에 숫자를 더하면 더한 숫자 만큼의 다음 알파벳 문자를 의미한다. 'K'에서 다음 4번째 문자(K, L, M, N, O)는 O이므로 O를 출력한다.

결과 1. KOREA
2. OREA
3. K
4. E
5. O

[문제 4]

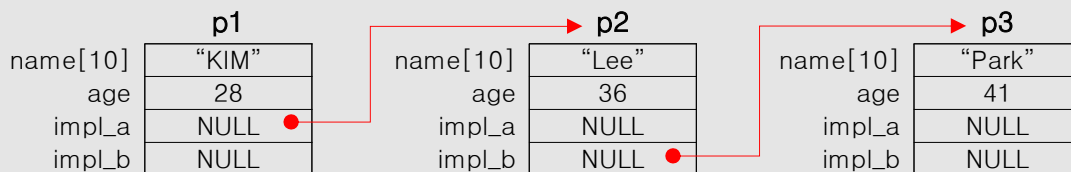
->

[해설]

```
#include <stdio.h>
struct insa {
    char name[10];
    int age;
    struct insa* impl_a;
    struct insa* impl_b;
};

main() {
    ① struct insa p1 = { "Kim", 28, NULL, NULL };
    ② struct insa p2 = { "Lee", 36, NULL, NULL };
    ③ struct insa p3 = { "Park", 41, NULL, NULL };
    ④ p1.impl_a = &p2;
    ⑤ p2.impl_b = &p3;
    ⑥ printf("%s\n", p1.impl_a->name);
    ⑦ printf("%d", p2.impl_b->age);
}
```

①~③번의 구조체 생성 과정과 ④, ⑤번의 주소 저장 과정을 그림으로 표현하면 다음과 같다.



⑥ p1.impl_a가 가리키는 곳의 name에 저장된 문자열을 출력한 후 커서를 다음 줄의 처음으로 옮긴다. p1.impl_a는 p2를 가리키므로 p2.name이 출력된다.

결과 Lee

⑦ p2.impl_b가 가리키는 곳의 age에 저장된 값을 정수로 출력한다. p2.impl_b는 p3을 가리키므로 p3.age가

출력된다.

결과 Lee
41

[문제 5]

chmod, 751

[해설]

- UNIX에서는 파일의 권한(permission)을 10자리로 표현하는데 1번째 자리는 디렉터리(d) 또는 파일(-), 2~4번째 자리는 소유자(Owner) 권한을, 5~7번째 자리는 그룹(Group) 권한을, 8~10번째 자리는 기타 사용자(Other) 권한을 의미합니다.
- 각 자리는 r(읽기), w(쓰기), x(실행), -(권한없음)으로 표시합니다.
- 사용자는 rwx, 그룹은 r-x, 기타 사용자는 --x 권한을 부여하므로, 파일 구분을 제외한 각 권한을 권한있음(1)과 권한없음(0)으로 바꾼 뒤 8진수로 변환하여 chmod 명령어의 매개 변수로 사용하면 됩니다.
- 111 101 001 → 751 → chmod 751

[문제 6]

패키지

[문제 7]

Equivalence Partition

[문제 8]

A
4
3
2
1

[풀이]

SELECT A FROM R	<R> 테이블의 'A' 속성을 표시한다.
UNION	두 SELECT문의 조회 결과를 통합하되 중복된 행은 한 번만 출력한다.
SELECT A FROM S	<S> 테이블의 'A' 속성을 표시한다.
ORDER BY A DESC;	'A' 속성을 기준으로 내림차순 정렬한다.

[문제 9]

34

[해설]

```

#include <stdio.h>
④ int isPerfectNum(int num) {
⑤     int sum = 0;
⑥     for (int i = 1; i < num; i++) {
⑦         if (num % i == 0)
⑧             sum += i;
⑨     }
⑨     if (num == sum) return 1;
⑩     else return 0;
    }
    main() {
①         int r = 0;
②         for (int i = 1; i <= 100; i++)
③⑪             if (isPerfectNum(i))
⑫                 r += i;
⑬         printf("%d", r);
    }

```

모든 C 언어 프로그램은 반드시 main() 함수에서 시작한다.

- ① 정수형 변수 r을 선언하고 0으로 초기화한다.
- ② 반복 변수 i가 1부터 1씩 증가하면서 100보다 작거나 같은 동안 ③~⑫번을 반복 수행한다.
- ③ i의 값을 인수로 isPerfectNum() 함수를 호출한 후 돌려받은 값이 참(1)이면 ⑫번으로 이동하고, 아니면 반복문의 시작인 ②번으로 이동한다.
- ④ 정수를 반환하는 isPerfectNum() 함수의 시작점이다. ③번에서 전달받은 i의 값을 num이 받는다.
- ⑤ 정수형 변수 sum을 선언하고 0으로 초기화한다.
- ⑥ 반복 변수 i가 1부터 1씩 증가하면서 num보다 작은 동안 ⑦, ⑧번을 반복 수행한다.
- ⑦ num을 i로 나눈 나머지가 0이면 ⑧번으로 이동한다.
- ⑧ 'sum = sum + i;'와 동일하다. sum에 i의 값을 누적시킨다.
※ ⑥, ⑦번은 ③번에서 보낸 인수의 자기 수를 제외한 약수를 찾는 과정이며, ⑧번은 찾은 약수를 sum에 누적하는 실행문이다.
- ⑨ num과 sum의 값이 같으면 함수를 호출했던 ⑪번으로 1(참)을 반환하고, 같지 않으면 ⑩번으로 이동한다.
- ⑩ 함수를 호출했던 ⑪번으로 0(거짓)을 반환한다.
※ 자기를 제외한 약수의 합이 자기와 같은 수를 완전수(Perfect Number)라고 한다. ⑨, ⑩번은 인수가 완전수임을 확인하는 과정으로, 완전수이면 참(1)을, 완전수가 아니면 거짓(0)을 반환한다. (ex: 6의 약수는 1, 2, 3이며, 1+2+3은 6과 같으므로 6은 완전수이다.)
- ⑪ ⑨, ⑩번에서 돌려받은 값이 1(참)이면 ⑫번으로 이동하고, 0(거짓)이면 반복문의 시작인 ②번으로 이동한다.
- ⑫ 'r = r + i;'와 동일하다. r에 i의 값을 누적시킨다.
- ⑬ r의 값을 정수로 출력한다.

결과 34

※ ⑪~⑬번을 통해 r에 완전수를 누적하고, ②번을 통해 100까지 반복한다는 것을 알았으므로, 이 코드는 결국 1부터 100까지의 숫자 중 완전수인 수를 합하여 출력하는 것임을 알 수 있다.

반복문 실행에 따른 변수들의 변화는 다음과 같다.

main()		isPerfectNum()		
i	r	num	i	sum
1	0	1	1	0
2		2	1 2	0 1
3		3	1 2 3	0 1 1
⋮	⋮	⋮	⋮	⋮
6	6	6	1 2 3 4 5 6	0 1 3 6 10 15
⋮	⋮	⋮	⋮	⋮
28	34	28	1 2 3 ⋮ 27 28	0 1 3 7 14 28
⋮		⋮	⋮	⋮
100		100	1 2 3 ⋮ 99 100	0 1 3 7 12 ⋮ 117

[문제 10]

NAT

[해설]

NAT(Network Address Translation)

[문제 11]

※ 다음 중 하나를 쓰면 됩니다.

ATM, 비동기 전송 방식, Asynchronous Transfer Mode

[문제 12]

7

[풀이]

- main() 메소드가 아닌 다른 메소드에 접근하기 위해서는 클래스를 메모리에 할당하는, 즉 객체 변수를 선언하는 과정이 필요합니다.

- 반면 static으로 선언된 메소드는 객체 변수를 선언하지 않아도 클래스명을 사용해서 Person.get()과 같은 형태로 접근할 수 있습니다. 즉 static으로 선언된 메소드는 메모리에 클래스를 위한 공간이 할당되지 않았다는 것을 의미합니다. 그러므로 static으로 선언된 메소드에서 메모리에 존재하지도 않은 클래스의 변수 name을 참조하는 것은 불가능합니다.
- 15번 줄에서 객체 변수를 선언하였지만 오류가 발생하는 7번 줄이 참조하는 name은 객체 변수를 특정할 수 없으며, 오류가 발생할 수 있는 코드가 있으면 해당 코드는 컴파일되지 않고 오류를 반환합니다.

[문제 13]

- ① MAC ② RBAC ③ DAC

[문제 14]

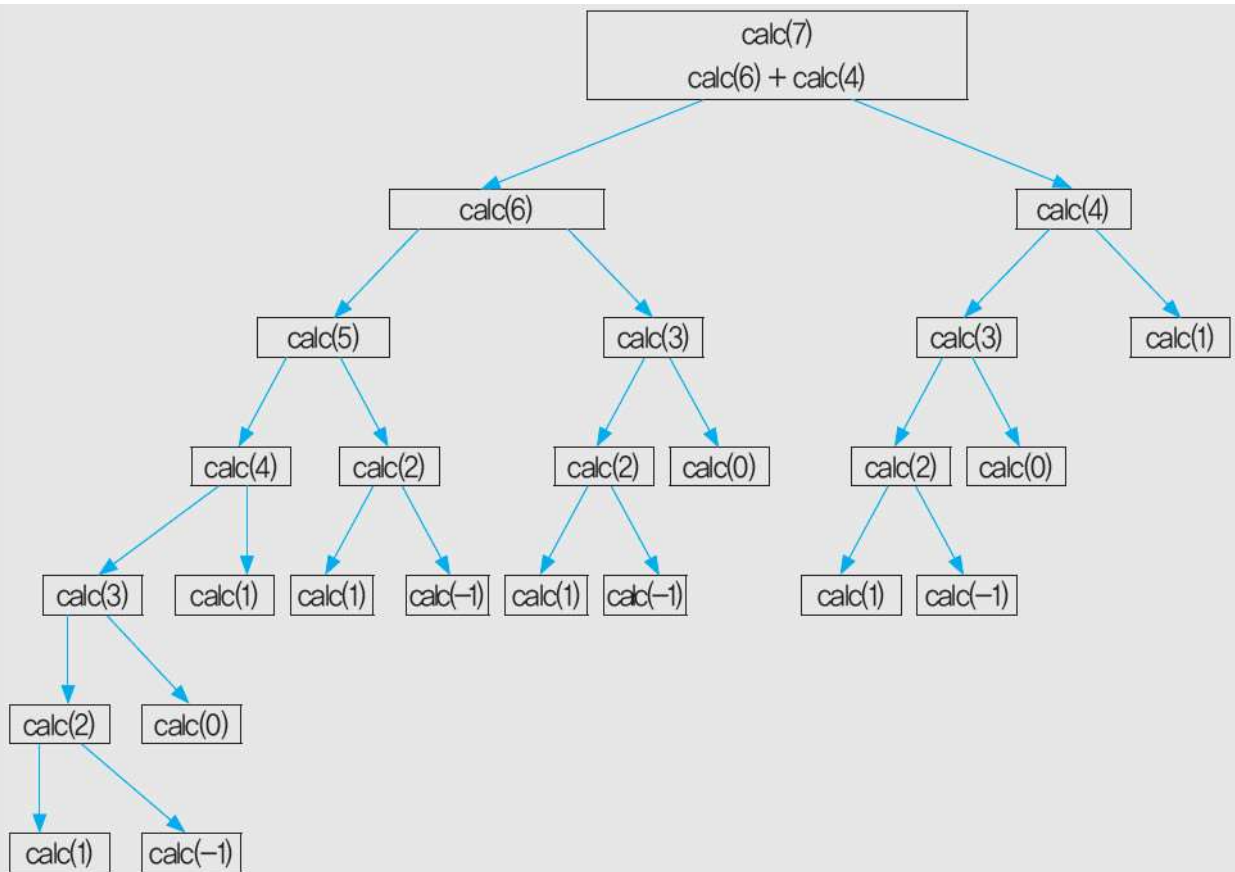
2

[해설]

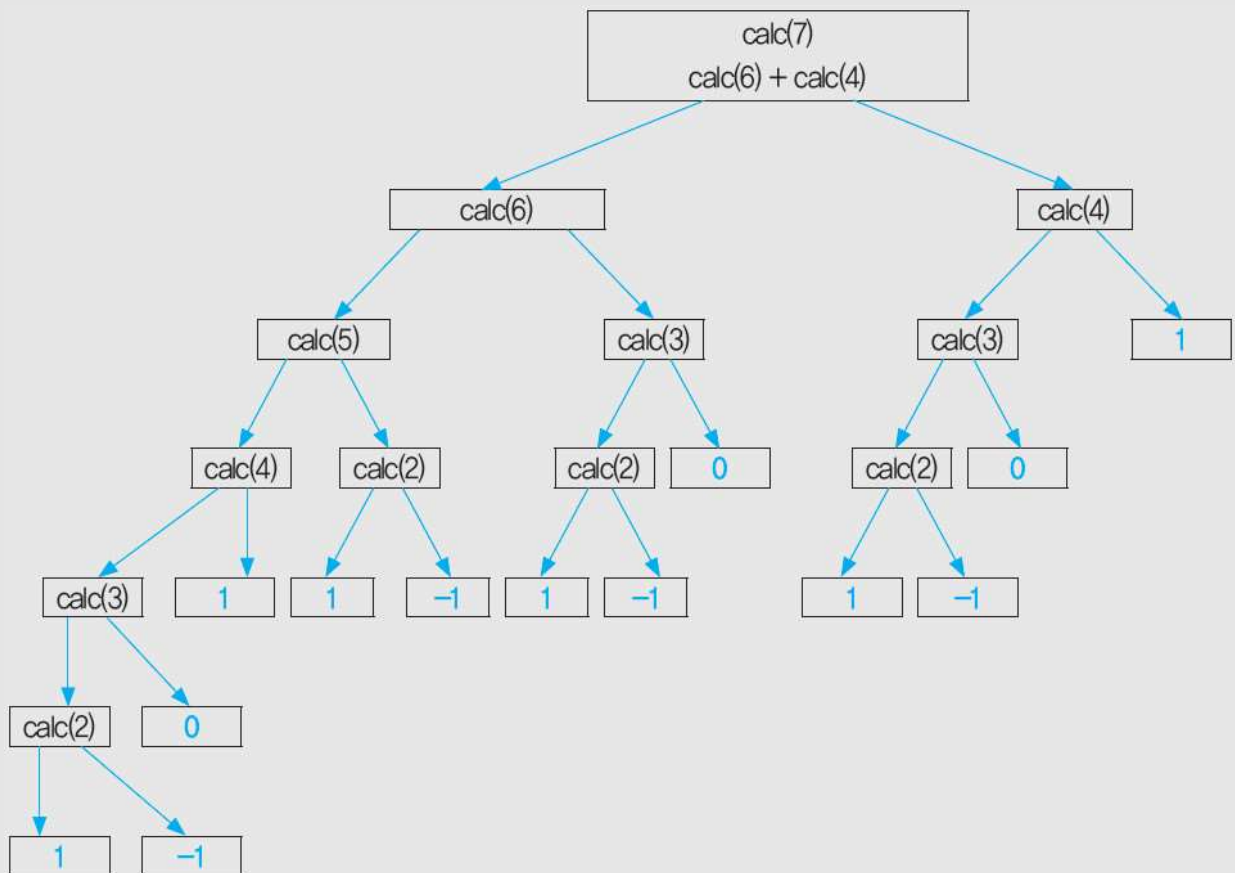
```
class P {
    public int calc(int n) {
        if (n <= 1) return n;
        return calc(n - 1) + calc(n - 2);
    }
}
class C extends P {
    ③    public int calc(int n) {
    ④        if (n <= 1) return n;
    ⑤        return calc(n - 1) + calc(n - 3);
    }
}
public class Test {
    public static void main(String[] args) {
    ①        P obj = new C();
    ②⑥        System.out.print(obj.calc(7));
    }
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

- ① C 클래스의 생성자를 이용하여 P 클래스의 객체 변수 obj를 생성한다.
- ② 7을 인수로 obj의 calc() 메소드를 호출한 후 돌려받은 값을 출력한다. ③번으로 이동한다.
obj.calc()는 obj 객체의 자료형이 P이므로 P.calc()라고 생각할 수 있지만 ①번에서 클래스 형 변환이 발생하였고, calc() 메소드가 자식 클래스에서 재정의되었으므로 C 클래스의 calc() 메소드가 호출된다.
- ③ 정수를 반환하는 calc() 메소드의 시작점이다. ②번에서 전달받은 7을 n이 받는다.
- ④ n이 1보다 작거나 같으면 n의 값을 반환하고, 아니면 ⑤번으로 이동한다.
- ⑤ n-1을 인수로 calc() 메소드를 호출한 후 돌려받은 값과 n-3을 인수로 calc() 메소드를 호출한 후 돌려받은 값을 합한 값을 함수를 호출했던 곳으로 반환한다.
• ③~⑤번에 해당하는 재귀호출문은 다음과 같이 표현할 수 있다.



· ④번에 의해 인수가 1 이하인 경우는 인수의 값을 그대로 반환하므로 이를 반영하면 다음과 같다.



· 값이 반환되어 적용되는 과정은 다음과 같다.

④ return n * f(n - 1);

n * f(n-1)을 연산한 후 함수를 호출했던 ①번으로 결과를 반환한다. n은 7이므로 f(6)을 호출한다.

```
⑤ int f(int n) {           //n은 6입니다.  
⑥     if (n <= 1) return 1;  
⑦     else return n * f(n - 1);  
    }
```

n은 6이므로 ⑦번을 수행한다. f(5)를 호출한다.

```
⑧ int f(int n) {           //n은 5입니다.  
⑨     if (n <= 1) return 1;  
⑩     else return n * f(n - 1);  
    }
```

n은 5이므로 ⑩번을 수행한다. f(4)를 호출한다.

```
⑪ int f(int n) {           //n은 4입니다.  
⑫     if (n <= 1) return 1;  
⑬     else return n * f(n - 1);  
    }
```

n은 4이므로 ⑬번을 수행한다. f(3)을 호출한다.

```
⑭ int f(int n) {           //n은 3입니다.  
⑮     if (n <= 1) return 1;  
⑯     else return n * f(n - 1);  
    }
```

n은 3이므로 ⑯번을 수행한다. f(2)를 호출한다.

```
⑰ int f(int n) {           //n은 2입니다.  
⑱     if (n <= 1) return 1;  
㉑     else return n * f(n - 1);  
    }
```

n은 2이므로 ㉑번을 수행한다. f(1)을 호출한다.

```
⑳ int f(int n) {           //n은 1입니다.  
㉒     if (n <= 1) return 1;  
    else return n * f(n - 1);  
    }
```

n이 1이므로 ㉒의 조건을 만족한다. 'return 1;'을 수행하여 함수의 실행을 종료하고 1을 반환하면서 제어를 f(1) 함수를 호출했던 ㉑번으로 옮긴다.

```
⑰ int f(int n) {           //n은 2입니다.  
⑱     if (n <= 1) return 1;  
⑲㉒     else return n * f(n - 1);  
    }
```

㉒ return n * f(n - 1);

㉑번에서 f(1)이 1을 반환하였으므로 2를 반환하면서 제어를 f(2) 함수를 호출했던 ㉑번으로 옮긴다.

return n * f(n - 1);

㉑ ㉒

• ㉑ : 2 ('f(n-1)'을 호출할 때 n은 2였으므로)

• ㉒ : 1 (㉑번에서 1을 반환하였으므로)

```
14 int f(int n) {           //n은 3입니다.
15     if (n <= 1) return 1;
16 ㉓     else return n * f(n - 1);
    }
```

㉓ return n * f(n - 1);

㉒번에서 f(2)가 2를 반환하였으므로 6을 반환하면서 제어를 f(3) 함수를 호출했던 ㉑번으로 옮긴다.

return n * f(n - 1);

3 2

```
11 int f(int n) {           //n은 4입니다.
12     if (n <= 1) return 1;
13 ㉔     else return n * f(n - 1);
    }
```

㉔ return n * f(n - 1);

㉓번에서 f(3)이 6을 반환하였으므로 24를 반환하면서 제어를 f(4) 함수를 호출했던 ㉑번으로 옮긴다.

return n * f(n - 1);

4 6

```
8 int f(int n) {           //n은 5입니다.
9     if (n <= 1) return 1;
10 ㉕     else return n * f(n - 1);
    }
```

㉕ return n * f(n - 1);

㉔번에서 f(4)가 24를 반환하였으므로 120을 반환하면서 제어를 f(5) 함수를 호출했던 ㉑번으로 옮긴다.

return n * f(n - 1);

5 24

```
5 int f(int n) {           //n은 6입니다.
6     if (n <= 1) return 1;
7 ㉖     else return n * f(n - 1);
    }
```

㉖ return n * f(n - 1);

㉕번에서 f(5)가 120을 반환하였으므로 720을 반환하면서 제어를 f(6) 함수를 호출했던 ㉑번으로 옮긴다.

return n * f(n - 1);

6 120

```

② int f(int n) {           //n은 7입니다.
③     if (n <= 1) return 1;
④②⑦ else return n * f(n - 1);
    }

```

②⑦ return n * f(n - 1);

②⑥번에서 f(6)이 720을 반환하였으므로 5040을 반환하면서 제어를 f(7) 함수를 호출했던 ①번으로 옮긴다.

```

return n * f(n - 1);
      7      720

```

```

main() {
①②⑥ printf("%d", f(7));
}

```

②⑥ ②⑦번에서 돌려받은 값 5040을 정수로 출력한다.

결과 **5040**

[문제 16]

split

[문제 17]

① IaaS ② PaaS ③ SaaS

[문제 18]

※ 다음 중 하나를 쓰면 됩니다.

RIP, 경로 선택 정보 프로토콜, Routing Information Protocol

[문제 19]

- ① Join : ▷◁
- ② Project : π
- ③ Select : σ
- ④ Division : ÷

[문제 20]

참조