



시험에 나오는것만 공부한다!

시나공시리즈

기출문제

2022년 1회 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 필기구만 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 준수 하시오.) (5점)

```
class A {
    int a;
    int b;
}

public class Test {
    static void func1(A m) {
        m.a *= 10;
    }
    static void func2(A m) {
        m.a += m.b;
    }
    public static void main(String args[]) {
        A m = new A();
        m.a = 100;
        func1(m);
        m.b = m.a;
        func2(m);
        System.out.printf("%d", m.a);
    }
}
```

답 :

문제 2 다음 설명의 RAID에 해당하는 레벨(Level)을 답 란의 괄호 안에 숫자로 쓰시오. (5점)

여러 개의 하드디스크로 디스크 배열을 구성하고, 파일을 구성하는 데이터 블록들을 서로 다른 디스크에 분산하여 저장할 경우 저장된 블록들을 여러 디스크에서 동시에 읽고 쓸 수 있으므로 디스크의 속도가 매우 향상되는데, 이 기술을 RAID라고 한다.

레이드 방식 중 패리티가 없는 스트리핑된 2개 이상의 디스크를 병렬로 연결하여 구성하는 이 방식은 디스크의 개수가 증가할수록 입·출력 속도 및 저장 용량이 배로 증가하지만, 하나의 디스크만 손상되어도 전체 데이터가 유실되는 문제가 발생한다.

답 : Level ()

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 3 다음 설명에 해당하는 DB의 트랜잭션 연산을 <보기>에서 찾아 쓰시오. (5점)

DBMS는 데이터베이스에 치명적인 손실이 발생했을 때 이를 복구하기 위해 데이터베이스의 처리 내용이나 이용 상황 등 상태 변화를 시간의 흐름에 따라 기록한 로그를 생성한다.

- (①) : 데이터베이스가 비정상적으로 종료되었을 때, 디스크에 저장된 로그를 분석하여 트랜잭션의 시작(start)과 완료(commit)에 대한 기록이 있는 트랜잭션들의 작업을 재작업한다. 즉 로그를 이용하여 해당 데이터 항목에 대해 이전 값을 이후 값으로 변경하는 연산이다.
- (②) : 데이터베이스가 비정상적으로 종료되었을 때, 디스크에 저장된 로그를 분석하여 트랜잭션의 시작을 나타내는 'start'는 있지만 완료를 나타내는 'commit' 기록이 없는 트랜잭션들이 작업한 내용들을 모두 취소한다. 즉 로그를 이용하여 해당 데이터 항목에 대해 이후 값을 이전 값으로 변경한다.

<보기>

<input type="radio"/> ROLLBACK	<input type="radio"/> UNDO	<input type="radio"/> LOG	<input type="radio"/> COMMIT
<input type="radio"/> REDO	<input type="radio"/> RECOVERY	<input type="radio"/> BACKUP	<input type="radio"/> CHECK

답

- ①
- ②

문제 4 다음은 <성적> 테이블에서 이름(name)과 점수(score)를 조회하되, 점수를 기준으로 내림차순 정렬하여 조회하는 <SQL문>이다. 괄호(①~③)에 알맞은 답을 적어 <SQL문>을 완성하시오. (5점)

<성적>

name	class	score
정기찬	A	85
이영호	C	74
환정형	C	95
김지수	A	90
최은영	B	82

<SQL문>

```
SELECT name, score
FROM 성적
( ① ) BY ( ② ) ( ③ )
```

답

- ①
- ②
- ③

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 5 다음 Java로 구현된 프로그램을 분석하여 괄호에 들어갈 알맞은 답을 쓰시오. (5점)

```
class Car implements Runnable {
    int a;
    public void run() {
        try {
            while(++a < 100) {
                System.out.println("miles traveled : " + a);
                Thread.sleep(100);
            }
        } catch(Exception E) { }
    }
}

public class Test {
    public static void main(String args[]) {
        Thread t1 = new Thread(new (      )());
        t1.start();
    }
}
```

답 :

문제 6 데이터의 중복으로 인해 테이블 조작 시 문제가 발생하는 현상을 이상(Anomaly)이라고 한다. 이상 중 삭제 이상(Deletion Anomaly)에 대해 간략히 서술하시오. (5점)

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 7 다음 Python로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
def func(num1, num2 = 2):  
    print('a =', num1, 'b =', num2)  
func(20)
```

답 :

문제 8 다음은 Python의 리스트 객체에 속한 메소드들에 대한 설명이다. 각 괄호(①~③)에 해당하는 메소드의 이름을 <보기>에서 찾아 쓰시오. (5점)

Python에서는 여러 요소들을 한 개의 이름으로 처리할 때 리스트(List)를 사용하며, 각 요소에는 정수, 실수, 문자열 등 다양한 자료형을 섞어 저장할 수 있다. 또한 리스트는 메소드를 활용하여 요소를 추가 및 삭제할 수 있을 뿐만 아니라 정렬하거나 다른 리스트와 병합하는 등의 다양한 작업을 손쉽게 수행할 수 있다.

- (①) : 기존 리스트에 인수의 요소들을 추가하여 확장하는 메소드로, 여러 값을 한 번에 추가할 수 있다.
- (②) : 리스트에서 맨 마지막 또는 인수의 값에 해당하는 위치의 요소를 삭제한 후 반환한다.
- (③) : 리스트에 저장된 각 요소들의 순서를 역순으로 뒤집어 저장하는 메소드이다.

<보기>

㉠ pop()	㉡ push()	㉢ reverse()	㉣ index()
㉤ write()	㉥ sort()	㉦ extend()	㉧ copy()

답

- ①
- ②
- ③

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 9 보안 프로토콜에 대한 다음 설명에 해당하는 용어를 영문 약어로 쓰시오. (5점)

무선랜 보안에 사용된 웹 방식을 보완한 데이터 보안 프로토콜로, 임시 키 무결성 프로토콜이라고도 한다. WEP의 취약성을 보완하기 위해 암호 알고리즘의 입력 키 길이를 128비트로 늘리고 패킷당 키 할당, 키값 재설정 등 키 관리 방식을 개선하였다.

답 :

문제 10 소스 코드 품질 분석 도구에 대한 다음 설명에서 괄호(①, ②)에 해당하는 용어를 <보기>에서 찾아 쓰시오. (5점)

소스 코드 품질 분석 도구는 소스 코드의 코딩 스타일, 코드에 설정된 코딩 표준, 코드의 복잡도, 코드에 존재하는 메모리 누수 현상, 스레드 결함 등을 발견하기 위해 사용하는 분석 도구이다.

- (①) 도구는 작성한 소스 코드를 실행하지 않고 코딩 표준이나 코딩 스타일, 결함 등을 확인하는 코드 분석 도구이다.
- (②) 도구는 소스 코드를 직접 실행하여 프로그램의 동작이나 반응을 추적하고 보고하는 분석 도구로, 프로그램 모니터링 기능이나 스냅샷 생성 기능들을 포함하고 있다.

<보기>

- | | | | |
|--------------------|--------------------|------------------|-------------------|
| ㉠ Static Analysis | ㉡ Running Analysis | ㉢ Test Execution | ㉣ Performance |
| ㉤ Dynamic Analysis | ㉥ Test Control | ㉦ Test Harness | ㉧ Test Monitoring |

답

- ①
- ②

문제 11 인터페이스 구현 검증 도구에 대한 다음 설명에 해당하는 용어를 영문으로 쓰시오. (5점)

Kent Beck과 Erich Gamma 등이 개발한 자바 프로그래밍 언어용 유닛 테스트 프레임워크로, xUnit 계열의 한 종류다. 같은 테스트 코드를 여러 번 작성하지 않게 도와주며, 테스트마다 예상 결과를 기억할 필요가 없는 자동화된 해법을 제공한다는 특징이 있다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 12 다음 C언어로 구현된 프로그램을 분석하여 5를 입력했을 때 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
int func(int a) {
    if (a <= 1) return 1;
    return a * func(a - 1);
}
int main() {
    int a;
    scanf("%d", &a);
    printf("%d", func(a));
}
```

답 :

문제 13 사용자 인터페이스(UI)에 대한 다음 설명에 해당하는 용어를 영문 약어로 쓰시오. (5점)

사용자의 자연스러운 움직임을 통해 시스템과 상호작용하는 사용자 인터페이스(UI)로, 키보드나 마우스와 같이 조작을 배워야 하는 인공 제어 장치를 사용하는 인터페이스와 구분하기 위해 ‘자연스러운’이라는 표현을 사용한다. 시리(Siri), 빅스비(Bixby) 등과 같은 음성 비서에게 사용하는 자연어 명령어나 휴대폰이나 태블릿에서의 터치 등이 여기에 해당한다.

답 :

문제 14 다음 중 블랙 박스 테스트 기법에 해당하는 것을 모두 골라 기호(㉠~㉨)로 쓰시오. (5점)

- | | | |
|----------------------------|------------------------------|------------------------------|
| ㉠ Base Path Testing | ㉡ Condition Testing | ㉢ Boundary Value Analysis |
| ㉣ Equivalence Partitioning | ㉤ Data Flow Testing | ㉥ Cause-Effect Graph |
| ㉦ Branch Coverage Testing | ㉧ Statement Coverage Testing | ㉨ Boundary Division Analysis |

답 :

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

연 습 란

문제 15 다음은 정수를 역순으로 출력하는 C언어 프로그램이다. 예를 들어 1234의 역순은 4321이다. 단, 1230 처럼 0으로 끝나는 정수는 고려하지 않는다. 프로그램을 분석하여 괄호(①~③)에 들어갈 알맞은 연산자를 쓰시오. (5점)

```
#include <stdio.h>
int main() {
    int number = 1234;
    int div = 10, result = 0;

    while (number ( ① ) 0) {
        result = result * div;
        result = result + number ( ② ) div;
        number = number ( ③ ) div;
    }
    printf("%d", result);
}
```

답

- ①
- ②
- ③

문제 16 정보 보호에 대한 다음 설명에 해당하는 용어를 영문 약어로 쓰시오. (5점)

정보 자산을 안전하게 보호하기 위한 보호 절차와 대책으로, 정보보호 관리 체계라고 한다. 조직에 맞는 정보보호 정책을 수립하고, 위험에 상시 대응하는 여러 보안 대책을 통합 관리한다. 공공 부문과 민간 기업 부문에서 이것을 평가하고 인증하는 사업을 한국인터넷진흥원(KISA)에서 운영중이다.

답 :

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 17 다음 C언어로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
int isPrime(int number) {
    for (int i = 2; i < number; i++)
        if (number % i == 0) return 0;
    return 1;
}

int main() {
    int number = 13195;
    int max_div = 0;
    for (int i = 2; i < number; i++)
        if (isPrime(i) == 1 && number % i == 0) max_div = i;
    printf("%d", max_div);
}
```

답 :

문제 18 키(Key)에 대한 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 용어를 쓰시오. (5점)

키(Key)는 데이터베이스에서 조건에 만족하는 튜플을 찾거나 순서대로 정렬할 때 기준이 되는 속성을 말한다.

- 슈퍼키(Super Key)는 한 릴레이션 내에 있는 속성들의 집합으로 구성된 키로, 릴레이션을 구성하는 모든 튜플에 대해 (①)을 만족한다.
- 후보키(Candidate Key)는 릴레이션을 구성하는 속성들 중에서 튜플을 유일하게 식별하기 위해 사용되는 속성들의 부분집합으로, (①)과 (②)을 만족하는 특징이 있다.

답

- ①
- ②

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

문제 19 보안 위협에 대한 다음 설명에 해당하는 용어를 <보기>에서 찾아 쓰시오. (5점)

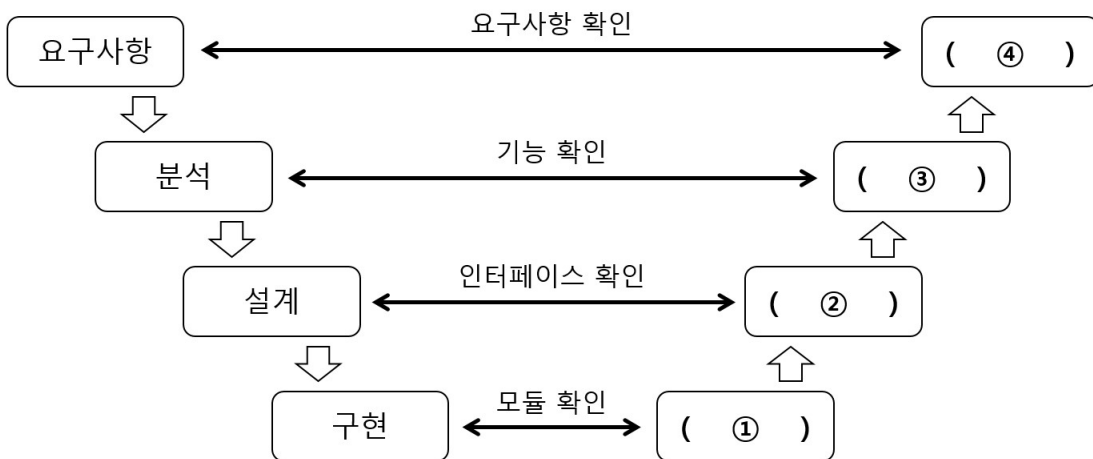
목표 조직이 자주 방문하는 웹 사이트를 사전에 감염시켜 목표 조직의 일원이 웹 사이트에 방문했을 때 악성 코드에 감염되게 한다. 이후에는 감염된 PC를 기반으로 조직의 중요 시스템에 접근하거나 불능으로 만드는 등의 영향력을 행사하는 웹 기반 공격이다.

<보기>

- | | | | |
|------------|---------------------|-----------------|-----------------|
| ㉠ Pharming | ㉡ Drive-by Download | ㉢ Watering Hole | ㉣ Business SCAM |
| ㉤ Phishing | ㉥ Cyber Kill Chain | ㉦ Ransomware | ㉧ Sniffing |

답 :

문제 20 개발 단계에 따른 애플리케이션 테스트에 대한 다음 V-모델에서 괄호(①~④)에 들어갈 알맞은 테스트를 쓰시오. (5점)



답

- ①
- ②
- ③
- ④

연 습 란

※ 다음 여백은 연습란으로 사용하시기 바랍니다.

기출문제 정답 및 해설

[문제 1]

2000

[해설]

```
class A {                                클래스 A를 정의한다.
    int a;                               클래스 A에는 정수형 변수 a와 b가 선언되어 있다.
    int b;
}

public class Test {
    ④ static void func1(A m) {
    ⑤     m.a *= 10;
    }
    ⑧ static void func2(A m) {
    ⑨     m.a += m.b;
    }

    public static void main(String args[]) {
    ①     A m = new A();
    ②     m.a = 100;
    ③     func1(m);
    ⑥     m.b = m.a;
    ⑦     func2(m);
    ⑩     System.out.printf("%d", m.a);
    }
}
```

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

① 클래스 A의 객체 변수 m을 선언한다.

	int a	int b
객체 변수 m		

② 객체 변수 m의 변수 a에 100을 저장한다.

	int a	int b
객체 변수 m	100	

③ 객체 변수 m의 시작 주소를 인수로 하여 func1 메소드를 호출한다.

④ 반환값이 없는 func1() 메소드의 시작점이다. ③번에서 전달받은 주소는 m이 받는다.

※ 객체 변수나 배열의 이름은 객체 변수나 배열의 시작 주소를 가리키므로, 인수로 전달하는 경우 메소드에서 변경된 값이 main()의 객체 변수나 배열에도 적용된다는 점을 염두에 두세요.

⑤ 'm.a = m.a * 10;'과 동일하다. m.a에 10을 곱한 값을 m.a에 저장한다. 메소드가 종료되었으므로 메소드를 호출했던 ③번의 다음 줄인 ⑥번으로 이동한다.

	int a	int b
객체 변수 m	1000	

⑥ m.b에 m.a의 값 1000을 저장한다.

	int a	int b
객체 변수 m	1000	1000

⑦ 객체 변수 m의 시작 주소를 인수로 하여 func2 메소드를 호출한다.

⑧ 반환값이 없는 func2() 메소드의 시작점이다. ⑦번에서 전달받은 주소는 m이 받는다.

⑨ 'm.a = m.a + m.b;'와 동일하다. m.a와 m.b를 합한 값을 m.a에 저장한다. 메소드가 종료되었으므로 메소드를 호출했던 ⑦번의 다음 줄인 ⑩번으로 이동한다.

	int a	int b
객체 변수 m	2000	1000

⑩ m.a의 값 2000을 정수로 출력한다.

결과 2000

[문제 2]

0

[문제 3]

① REDO ② UNDO

[문제 4]

① ORDER ② score ③ DESC

[답안 작성 시 주의 사항]

SQL에 사용되는 예약어, 필드명, 변수명 등은 대소문자를 구분하지 않기 때문에 대문자로만 또는 소문자로만 작성해도 정답으로 인정됩니다.

[해설]

SELECT name, score	'name'과 'score'를 표시한다.
FROM 성적	<성적> 테이블에서 검색한다.
ORDER BY score DESC	'score'를 기준으로 내림차순 정렬한다.

[문제 5]

Car

[답안 작성 시 주의 사항]

C, Java, Python 등의 프로그래밍 언어에서는 대소문자를 구분하기 때문에 클래스명도 대소문자를 구분하여 정확하게 작성해야 합니다. 예를 들어, 소문자로 car로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

```
① class Car implements Runnable {
    int a;
② public void run() {
③ ① try {
④ ② while(++a < 100) {
⑤ ③ System.out.println("miles traveled : " + a);
⑥ ④ Thread.sleep(100);
⑦ }
⑧ ⑤ } catch(Exception E) { }
⑨ }
⑩ }

public class Test {
    public static void main(String args[]) {
⑪ ⑥ Thread t1 = new Thread(new Car());
⑫ ⑦ t1.start();
}
```

```
} ③
}
```

④ class Car implements Runnable

Runnable 인터페이스를 상속받은 클래스 Car를 정의한다.

- **implements** : extends와 같이 상속에 사용하는 예약어로, 인터페이스를 상속받을 때 사용함
- **Runnable** : 스레드 클래스를 만들 때 사용하는 인터페이스

※ 인터페이스 개체는 클래스와 크게 다르지 않습니다. 그 역할이 인터페이스로 고정되어 있을 뿐 클래스와 마찬가지로 변수와 메소드를 갖는 개체입니다.

※ 스레드는 시스템의 여러 자원을 할당받아 실행하는 프로그램의 단위입니다. 대부분은 main() 메소드로 실행하는 하나의 스레드로만 작업을 수행하는데, 스레드 클래스는 main() 메소드로 실행하는 스레드 외에 추가적인 스레드를 가질 수 있도록 스레드를 생성하는 기능을 갖고 있습니다.

⑤ public void run()

Runnable 인터페이스를 상속받았다면 스레드가 수행할 작업들을 정의하는 run() 메소드를 반드시 정의해야 한다.

⑥ try { }

- 실행 중에 예외가 발생할 가능성이 있는 실행 코드들을 하나의 블록으로 묶어 놓은 곳이다. try 블록 코드를 수행하다 예외가 발생하면 예외를 처리하는 ⑦의 catch 블록으로 이동하여 예외 처리 코드를 수행하므로 예외가 발생한 이후의 코드는 실행되지 않는다.

- ④번에서 수행되는 Thread.sleep() 메소드는 인터럽트로 인한 예외를 발생시킬 가능성이 큰 메소드이므로 반드시 try ~ catch 문을 통해 예외를 처리해줘야 한다.

⑦ catch(Exception E) { }

인터럽트로 인한 예외를 처리할 수 있는 예외 객체는 InterruptedException이지만, Exception을 사용하면 InterruptedException을 포함한 대부분의 예외를 한 번에 처리할 수 있다.

⑧ Thread t1 = new Thread(new Car());

스레드 클래스의 객체 변수 t1을 선언한다. 스레드 클래스는 생성자를 호출할 때 Runnable 인터페이스를 인수로 사용한다. 여기에서는 Runnable 인터페이스를 상속받은 Car 클래스를 생성자의 인수로 사용했다.

⑨ t1.start();

t1의 start() 메소드를 호출한다. start() 메소드는 스레드 클래스에 포함된 메소드로, run() 메소드에서 정의한 코드들을 실행하는 메소드이다. 이때 run() 메소드에서 정의한 코드들은 main() 메소드와는 별개로 시스템으로부터 자원을 새로 할당받아 실행된다. 즉 main() 메소드와 별개로 실행되기 때문에 main() 메소드의 작업이 종료되어도 run() 메소드의 작업이 끝나지 않으면 계속 수행한다.

모든 Java 프로그램은 반드시 main() 메소드에서 시작한다.

① 스레드 클래스의 객체 변수 t1을 선언한다. 스레드에서 실행할 run() 메소드를 정의하고 있는 Car() 클래스를 생성자의 인수로 사용한다.

② t1의 start() 메소드를 호출한다. Car 클래스의 run() 메소드가 실행된다. 이후 main() 메소드는 ③번으로 이동하여 프로그램을 종료한다.

main() 메소드와는 별개로 시스템으로부터 자원을 새로 할당받아 run() 메소드를 시작한다.

① 예외를 처리하기 위한 try ~ catch문의 시작점이다.

② a가 100보다 작은 동안 ③, ④번을 반복 수행한다. a는 전치증가 연산이므로 a에 1을 더한 후 조건을 확인한다.

※ 클래스의 속성으로 선언된 변수 a는 자동으로 0으로 초기화됩니다.

③ miles traveled : 를 출력한 후 이어서 a의 값을 출력한다.

④ 100을 인수로 Thread 클래스의 sleep() 메소드를 호출한다. 0.1초 동안 스레드를 일시 정지시킨다.

- Thread.sleep(n) : n/1000초 동안 스레드를 일시 정지시킨다.

②~④번을 수행한 결과로 다음과 같이 0.1초마다 한 줄씩 출력된다.

결과

miles traveled : 1
miles traveled : 2
miles traveled : 3
miles traveled : 4
:
miles traveled : 98
miles traveled : 99

[문제 6]

※ 다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

테이블에서 튜플을 삭제할 때 의도와는 상관없는 값들도 함께 삭제되는 현상이다.

[문제 7]

a = 20 b = 2

[답안 작성 시 주의 사항]

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어, 출력값 사이에 쉼표를 넣어 a = 20, b = 2로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

[해설]

② def func(num1, num2 = 2):	def는 Python에서 메소드를 정의하는 예약어이다. 매개 변수를 지정 한 후 메소드에 속한 실행 코드들은 콜론(:)과 여백으로 구분한다.
③ print('a =', num1, 'b =', num2)	
① func(20)	

func 메소드를 정의하는 부분의 다음 줄인 3번째 줄부터 실행한다.

① 20을 인수로 func() 메소드를 호출한다.

② func() 메소드의 시작점이다. ①번에서 전달받은 20을 num1이 받는다.

- func() 메소드의 매개 변수는 num1, num2 두 개지만 num2는 메소드 정의 시 초기값이 지정되었다.
- 전달된 인수는 매개 변수에 차례로 전달되므로 인수가 하나만 주어지면 num1이 인수를 전달받고, 두 개의 인수가 주어지면 num1과 num2가 차례로 인수를 전달받는다.

③ a =와 num1의 값 20, b =와 num2의 값 2를 차례대로 출력한다.

결과 a = 20 b = 2

[문제 8]

① extend() ② pop() ③ reverse()

[문제 9]

TKIP

[문제 10]

① Static Analysis ② Dynamic Analysis

[문제 11]

JUnit

[문제 12]

120

[해설]

```

int main() {
❶   int a;
❷   scanf("%d", &a);
❸   printf("%d", func(a));
}

```

모든 C언어 프로그램은 반드시 main() 함수에서 시작한다.

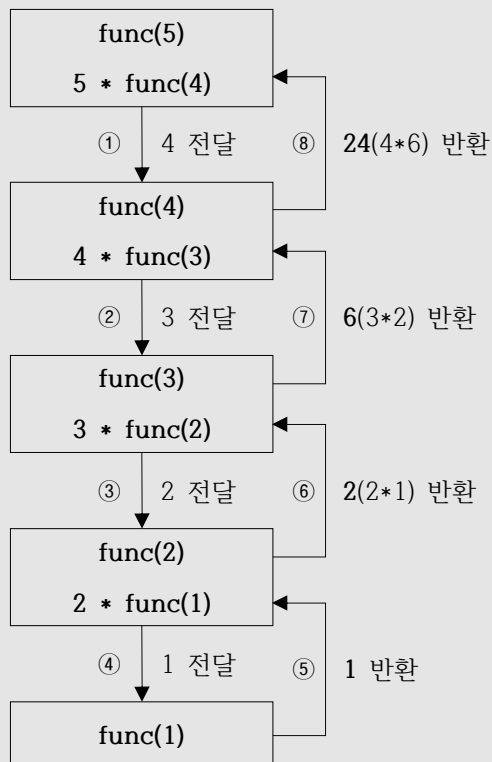
- ❶ 정수형 변수 a를 선언한다.
- ❷ 정수를 입력받아 a에 저장한다. 5가 입력되었다고 가정하였으므로 a에는 5가 저장된다.
- ❸ a의 값 5를 인수로 하여 func() 함수를 호출한 후 돌려받은 값을 정수로 출력한다.

```

❹ int func(int a) {
❺   if (a <= 1) return 1;
❻   return a * func(a - 1);
}

```

- ❹ 정수를 반환하는 func() 함수의 시작점이다. ❸번에서 전달받은 5를 a가 받는다.
- ❺ a가 1보다 작거나 같으면 함수를 호출했던 곳으로 1을 반환하고 함수를 종료한다. a의 값 5는 1보다 작거나 같지 않으므로 ❻번으로 이동한다.
- ❻ a-1을 인수로 하여 func() 함수를 호출한 후 돌려받은 값과 a를 곱하여 함수를 호출했던 곳으로 반환하고 함수를 종료한다. a가 1보다 큰 동안 자신을 호출하는 과정이 수행되다 a가 1보다 작거나 같아지면 1이 반환되면서 호출했던 과정을 복귀한다. 이 때 반환된 값은 먼저 호출된 func() 함수에 반환할 값으로 계산된다는 것을 염두에 두고 과정을 개괄적인 그림을 통해 살펴보자.



a-1의 값 4를 인수로 func() 함수를 호출한 후 돌려받은 값이 24이므로 a의 값 5를 곱한 값 120을 func(5)를 호출했던 ❸번으로 반환한다.

```

int main() {
  ❶ int a;
  ❷ scanf("%d", &a);
  ❸ ❷ printf("%d", func(a));
}

```

❷ ❸번에서 돌려받은 값 120을 정수로 출력한다.

결과 120

[문제 13]

NUI

[문제 14]

㉠, ㉡, ㉢

[문제 15]

❶ != 또는 > ❷ % ❸ /

[해설]

```

#include <stdio.h>
int main() {
  ❶ int number = 1234;
  ❷ int div = 10, result = 0;

  ❸ while (number != 0) {
    ❹ result = result * div;
    ❺ result = result + number % div;
    ❻ number = number / div;
  }
  ❼ printf("%d", result);
}

```

- ❶ 정수형 변수 number를 선언하고 1234로 초기화한다.
 - ❷ 정수형 변수 div와 result를 선언하고 각각 10과 0으로 초기화한다.
 - ❸ number가 0이 아닌 동안 ❹~❻번을 반복 수행한다.
 - ❹ result와 div를 곱한 값을 result에 저장한다.
 - ❺ number를 div로 나눈 나머지를 result에 누적시킨다.
 - ❻ number를 div로 나눈 값을 number에 저장한다.
- 반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

number	div	result
1234	10	0
		0
		4
123		
		40
		43
12		
		430
		432
1		
		4320
		4321
0		

⑦ result의 값 4321을 정수로 출력한다.

결과 4321

[문제 16]

ISMS

[문제 17]

29

[해설]

```
#include <stdio.h>
⑤ int isPrime(int number) {
⑥     for (int i = 2; i < number; i++)
⑦         if (number % i == 0) return 0;
⑧     return 1;
}

int main() {
①     int number = 13195;
②     int max_div = 0;
③     for (int i = 2; i < number; i++)
④⑨     if (isPrime(i) == 1 && number % i == 0) max_div = i;
⑩     printf("%d", max_div);
}
```

모든 C언어 프로그램은 반드시 main() 함수에서 시작한다.

- ① 정수형 변수 number를 선언하고 13195로 초기화한다.
- ② 정수형 변수 max_div를 선언하고 0으로 초기화한다.
- ③ 반복 변수 i가 2부터 1씩 증가하면서 number보다 작은 동안 ④번을 반복 수행한다.

첫 번째 반복

- ④ i의 값 2를 인수로 isPrime을 호출한 결과가 1이고 number를 i로 나눈 나머지가 0이면 max_div에 i의 값을 저장한다.
- ⑤ 정수를 반환하는 isPrime() 함수의 시작점이다. ④번에서 전달받은 2를 number가 받는다.
- ⑥ 반복 변수 i가 2부터 1씩 증가하면서 number보다 작은 동안 ⑦번을 반복 수행한다. i의 값 2는 number의

값 2보다 작지 않으므로 ⑦번을 수행하지 않고 ⑧번으로 이동한다.

⑧ 1을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑨ ⑧번에서 돌려받은 값은 1이지만, number의 값 13195를 i의 값 2로 나눈 나머지는 1이므로 `max_div = i`를 수행하지 않고 ③번으로 돌아가 i의 값을 1 증가시킨다.

두 번째 반복

④ i의 값 3을 인수로 `isPrime`을 호출한 결과가 1이고 number를 i로 나눈 나머지가 0이면 `max_div`에 i의 값을 저장한다.

⑤ 정수를 반환하는 `isPrime()` 함수의 시작점이다. ④번에서 전달받은 3을 number가 받는다.

⑥ 반복 변수 i가 2부터 1씩 증가하면서 number보다 작은 동안 ⑦번을 반복 수행한다.

⑦ number를 i로 나눈 나머지가 0이면 0을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑥~⑦번 반복문 실행에 따른 변수들의 변화는 다음과 같다.

number	i
3	2
	3

⑧ 1을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑨ ⑧번에서 돌려받은 값은 1이지만, number의 값 13195를 i의 값 3으로 나눈 나머지는 1이므로 `max_div = i`를 수행하지 않고 ③번으로 돌아가 i의 값을 1 증가시킨다.

세 번째 반복

④ i의 값 4를 인수로 `isPrime`을 호출한 결과가 1이고 number를 i로 나눈 나머지가 0이면 `max_div`에 i의 값을 저장한다.

⑤ 정수를 반환하는 `isPrime()` 함수의 시작점이다. ④번에서 전달받은 4를 number가 받는다.

⑥ 반복 변수 i가 2부터 1씩 증가하면서 number보다 작은 동안 ⑦번을 반복 수행한다.

⑦ number를 i로 나눈 나머지가 0이면 0을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑥~⑦번 반복문 실행에 따른 변수들의 변화는 다음과 같다.

number	i
4	2

⑨ ⑧번에서 돌려받은 값이 0이고, number의 값 13195를 i의 값 4로 나눈 나머지는 3이므로 `max_div = i`를 수행하지 않고 ③번으로 돌아가 i의 값을 1 증가시킨다.

네 번째 반복

④ i의 값 5를 인수로 `isPrime`을 호출한 결과가 1이고 number를 i로 나눈 나머지가 0이면 `max_div`에 i의 값을 저장한다.

⑤ 정수를 반환하는 `isPrime()` 함수의 시작점이다. ④번에서 전달받은 5를 number가 받는다.

⑥ 반복 변수 i가 2부터 1씩 증가하면서 number보다 작은 동안 ⑦번을 반복 수행한다.

⑦ number를 i로 나눈 나머지가 0이면 0을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑥~⑦번 반복문 실행에 따른 변수들의 변화는 다음과 같다.

number	i
5	2
	3
	4
	5

⑧ 1을 반환하고 함수를 호출했던 ⑨번으로 이동한다.

⑨ ⑧번에서 돌려받은 값이 1이고, number의 값 13195를 i의 값 5로 나눈 나머지도 0이므로 `max_div`에 5를 저장한 후 ③번으로 돌아가 i의 값을 1 증가시킨다.

⋮

위의 과정을 통해 다음 사항들을 알 수 있다.

- isPrime() 함수는 인수를 2에서 시작하여 전달받은 수보다 1 작을 때까지 나눴을 때 끝까지 나머지가 0이 아니면 1을 반환하는 것으로 보아 소수를 찾는 함수임을 알 수 있다.
- ⑨번에서 isPrime(i)가 1이라는 것은 i가 소수임을 의미하고, number를 i로 나눈 나머지가 0이라는 것은 i가 number의 약수라는 의미이므로, max_div에는 소수이자 number의 약수인 수가 저장된다.
- i의 값이 1씩 증가하면서 number보다 1 작을 때까지 위 과정을 수행하므로 number의 약수 중 가장 큰 소수에 해당하는 값이 max_div에 저장된다.
- 13195의 약수는 $5 \times 7 \times 13 \times 29$ 로 표현할 수 있으므로, 이 중 가장 큰 소수인 29가 최종적으로 max_div에 저장된다.
- 자세한 값의 변화는 다음 표를 통해 확인하자

main() 함수			isPrime() 함수		
number	i	max_div	number	i	반환값
13195 : :	2	0	2	2	1
	3		3	2	1
				3	
	4		4	2	0
	5	5	5	2	1
				3	
				4	
				5	
	:	:	:	:	:
	29	29	29	2	1
				3	
				4	
				:	
				28	
				29	
	:	:	:	:	:
	13194		13194	2	0
	13195				

⑩ max_div의 값 29를 정수로 출력한다.

결과 **29**

[문제 18]

※ 각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① 유일성, Unique
- ② 최소성, Minimality

[문제 19]

Watering Hole

[문제 20]

※ 각 문항별로 다음 중 하나를 쓰면 됩니다.

- ① 단위 테스트, Unit Test
- ② 통합 테스트, Integration Test
- ③ 시스템 테스트, System Test
- ④ 인수 테스트, Acceptance Test