



2020년 2회
정보처리기사 실기 시험
100% 합격전략집

최종점검 모의고사



문제 1 누구나 별다른 제한 없이 사용할 수 있도록 소스 코드가 공개되어 무료로 사용이 가능하고, 사용 시 라이선스의 종류, 사용자 수, 기술의 지속 가능성 등을 고려해야 하는 소프트웨어를 가리키는 용어를 쓰시오. (5점)

답 :

문제 2 요구사항 분석 기법에 대한 다음 설명에서 괄호에 공통으로 들어갈 알맞은 용어를 쓰시오. (5점)

요구사항 분석에서 자료의 흐름 및 변환 과정과 기능을 도형 중심으로 기술하는 방법을 () 또는 버블 차트라고 한다. ()의 각 구성 요소의 표기법은 다음과 같다.

프로세스 (Process)	<ul style="list-style-type: none"> • 자료를 변환시키는 시스템의 한 부분(처리 과정)을 나타내며 처리, 기능, 변환, 버블이라고한다. • 원이나 둥근 사각형으로 표시하고 그 안에 프로세스 이름을 기입한다.
자료 흐름 (Data Flow)	<ul style="list-style-type: none"> • 자료의 이동(흐름)이나 연관관계를 나타낸다. • 화살표 위에 자료의 이름을 기입한다.
자료 저장소 (Data Store)	<ul style="list-style-type: none"> • 시스템에서의 자료 저장소(파일, 데이터베이스)를 나타낸다. • 평행선 안에 자료 저장소 이름을 기입한다.
단말 (Terminator)	<ul style="list-style-type: none"> • 시스템과 교신하는 외부 개체로, 입력 데이터가 만들어지고 출력 데이터를 받는다(정보의 생산자와 소비자). • 사각형 안에 이름을 기입한다.

답 :

문제 3 반정규화(Denormalization)에 대한 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 용어를 쓰시오. (5점)

반정규화는 시스템의 성능 향상, 개발 및 운영의 편의성 등을 위해 정규화된 데이터 모델을 통합, 중복, 분리하는 과정으로, 의도적으로 정규화 원칙을 위배하는 행위이다. 이 중 데이터 모델을 분리하는 방법은 다음과 같이 2가지 방법으로 나눌 수 있다.

(①)	레코드(Record)를 기준으로 테이블을 분할하는 것으로, 레코드별로 사용 빈도의 차이가 큰 경우 사용 빈도에 따라 테이블을 분할한다.
(②)	하나의 테이블에 속성이 너무 많을 경우 속성을 기준으로 테이블을 분할한다.

답

- ① :
- ② :

문제 4 다음 C언어로 구현된 프로그램을 분석하여 10을 입력했을 때 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
main() {
    int a, b, k[10];
    scanf("%d", &a);
    b = aliquot(a, k);
    for (int j = 0; j < b; j++)
        printf("%d ", k[j]);
}
int aliquot(int a, int k[]) {
    int cnt = 0;
    for (int i = 1; i <= a; i++)
        if (a % i == 0) {
            k[cnt] = i;
            cnt++;
        }
    return cnt;
}
```

답 :

문제 5 테이블에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하고, 이 중복으로 인해 테이블 조작 시 문제가 발생하는 현상을 이상(Anomaly)라고 한다. 이러한 이상 중 테이블의 특정 튜플의 속성 값을 수정할 때 발생하는 이상으로, 일부 튜플의 정보만 갱신되어 정보에 불일치성(Inconsistency)이 생기는 현상을 가리키는 용어를 쓰시오. (5점)

답 :

문제 6 전사적으로 구축된 데이터 웨어하우스로부터 특정 주제나 부서 중심으로 구축된 소규모 단일 주제의 데이터 웨어하우스이다. 일반적으로 한 기업 내에 여러 개가 존재하여 특정 부서의 의사 결정을 지원하며, 다양한 질의나 요구를 충족하는 유연성과 접근성이 뛰어난 다차원 구조의 데이터를 지닌 이것을 가리키는 용어를 쓰시오. (5점)

답 :

문제 7 다음 <사원> 릴레이션을 참고하여 릴레이션의 차수(Degree)와 카디널리티(Cardinality)를 쓰시오. (5점)

<사원>

사번	이름	주민번호	부서	직위
1723511	차인환	790212-1*****	인사	부장
1822145	김진표	831212-1*****	회계	차장
1931521	서인영	830723-2*****	영업	차장
1922459	구준서	850507-1*****	개발	주임

답

- ① 차수(Degree) :
- ② 카디널리티(Cardinality) :

문제 8 EAI(Enterprise Application Integration)에 대한 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 용어를 쓰시오. (5점)

EAI는 기업 내 각종 애플리케이션 및 플랫폼 간의 정보 전달, 연계, 통합 등 상호 연동이 가능하게 해주는 솔루션으로, 비즈니스 간 통합 및 연계성을 증대시켜 효율성 및 각 시스템 간의 확정성(Determinacy)을 높여 준다.

EAI의 구축 유형은 다음과 같다.

Point-to-Point	<ul style="list-style-type: none"> • 가장 기본적인 애플리케이션 통합 방식으로, 애플리케이션을 1:1로 연결한다. • 변경 및 재사용이 어렵다.
(①)	<ul style="list-style-type: none"> • 단일 접점인 허브 시스템을 통해 데이터를 전송하는 중앙 집중형 방식이다. • 확장 및 유지 보수가 용이하며, 허브 장애 발생 시 시스템 전체에 영향을 미친다.
(②)	<ul style="list-style-type: none"> • 애플리케이션 사이에 미들웨어를 두어 처리하는 방식이다. • 확장성이 뛰어나며 대용량 처리가 가능하다.
Hybrid	<ul style="list-style-type: none"> • 그룹 내에서는 허브 시스템을 통해 데이터를 전송하고, 그룹 간에는 미들웨어를 두어 처리하는 방식이다. • 데이터 병목 현상을 최소화할 수 있다.

답

- ① :
- ② :

문제 9 다음 C언어로 구현된 프로그램과 <출력>을 분석하여 괄호에 들어갈 적합한 코드를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
#include <stdio.h>
main() {
    int a[5][5], 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;
    }

    for (int i = 0; i < 5; i++) {
        for (int j = 0; j < 5; j++)
            printf("%3d", a[i][j]);
        printf("\n");
    }
}
```

<출력>

```
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
```

답 :

문제 10 다양한 조건에 따라 SQL 구문을 유동적으로 변경하여 처리할 수 있으며, 애플리케이션 이용자로부터 SQL문의 일부 또는 전부를 입력받아 실행할 수 있는 SQL 처리 방식을 쓰시오. (5점)

답 :



문제 11 배치 프로그램에 대한 다음 설명에서 괄호(①, ②)에 들어갈 적합한 용어를 쓰시오. (5점)

배치 프로그램은 사용자와의 상호 작용 없이 여러 작업들을 미리 정해진 일련의 순서에 따라 일괄적으로 처리하는 것을 의미한다. 배치 스케줄러는 배치 프로그램 안에서 예약된 작업이 설정된 주기에 맞춰 자동으로 수행되도록 지원하는 도구로, 다음과 같은 종류가 있다.

스프링 배치 (Spring Batch)	<ul style="list-style-type: none"> • Spring Source 사와 Accenture 사가 2007년 공동 개발한 오픈소스 프레임워크이다. • 주요 구성 요소 : Job, Job Launcher, Step, Job Repository
(①)	<ul style="list-style-type: none"> • 스프링 프레임워크로 개발되는 응용 프로그램들의 일괄 처리를 위한 다양한 기능을 제공하는 오픈 소스 라이브러리이다. • 주요 구성 요소 : Scheduler, Job, JobDetail, Trigger
(②)	리눅스의 스케줄러 도구로 crontab 명령어를 통해 작업을 예약할 수 있다.

답

- ① :
- ② :

문제 12 UI 설계 도구에 대한 다음 설명에서 괄호에 공통으로 들어갈 적합한 답을 쓰시오. (5점)

- ()는 와이어프레임에 콘텐츠에 대한 설명, 페이지 간 이동 흐름 등을 추가한 문서이자 디자이너와 개발자가 최종적으로 참고하는 작업 지침서이다.
- ()에는 정책, 프로세스, 콘텐츠 구성, 와이어프레임, 기능 정의 등 서비스 구축을 위한 모든 정보가 들어 있다.
- ()의 우측에는 화면에 대한 설명, 전반적인 로직, 분기처리, 예외처리 등이 작성되는 부분이 있으며, 이를 디스크립션(Description)이라고 부른다.

답 :

문제 13 다음 Java로 구현된 프로그램을 분석하여 그 실행 결과를 쓰시오. (단, 출력문의 출력 서식을 준수하시오.) (5점)

```
public class Test {
    static void arr(int[] a) {
        int sw, temp, n = a.length;
        for(int i = 0; i < n - 1; i++) {
            sw = i;
            for(int j = i + 1; j < n; j++)
                if(a[j] < a[sw])
                    sw = j;
            temp = a[i];
            a[i] = a[sw];
            a[sw] = temp;
        }
    }
    public static void main(String[] args) {
        int n[] = { 4, 2, 7, 1 };
        arr(n);
        for(int i:n)
            System.out.printf("%d ", i);
    }
}
```

답 :

문제 14 애플리케이션 테스트에 대한 다음 설명에서 괄호(①, ②)에 들어갈 알맞은 테스트 종류를 쓰시오. (5점)

애플리케이션 테스트는 테스트 시 프로그램의 실행 여부 또는 진행 목적 등에 따라 다양하게 분류되는데, 이러한 분류 중 애플리케이션을 테스트 할 때 누구를 기준으로 하느냐에 따라 다음과 같이 분류할 수 있다.

(①)	개발자의 시각에서 제품의 생산 과정을 테스트하는 것으로, 제품이 명세서대로 완성됐는지를 테스트한다.
(②)	사용자의 시각에서 생산된 제품의 결과를 테스트하는 것으로, 사용자가 요구한대로 제품이 완성됐는지, 제품이 정상적으로 동작하는지를 테스트한다.

답

- ① :
- ② :



문제 15 구현된 소프트웨어가 사용자의 요구사항을 정확하게 준수했는지를 확인하기 위해 설계된 입력 값, 실행 조건, 기대 결과 등으로 구성된 테스트 항목에 대한 명세서로, 명세 기반 테스트의 설계 산출물에 해당하는 것은 무엇인지를 쓰시오. (5점)

답 :

문제 16 다음은 <학생> 테이블로부터 민감한 정보는 감추고 필요한 정보만을 노출하기 위해 <SQL문>을 실행하여 <학생_v> 뷰를 생성한 것이다. <SQL문>의 괄호(①, ②)에 들어갈 알맞은 명령어를 쓰시오.

<학생>

번호	이름	학과	성별	주소
2012	김영수	경제	남	경기
2103	박나래	경영	여	서울
2346	이미래	전자	여	인천



<학생_v>

번호	이름	학과
2012	김영수	경제
2103	박나래	경영
2346	이미래	전자

<SQL문>

CREATE (①) 학생_v (②) 번호, 이름, 학과 FROM 학생;

답

- ① :
- ② :

문제 17 서비스 거부 공격(DoS, Denial of Service)의 하나인 티어드롭 공격(Teardrop Attack)의 개념을 설명하시오. (5점)

답 :

문제 18 128비트의 블록 크기를 갖는 128비트 및 256비트 수준의 국산 대칭 키 블록 암호화 알고리즘으로, 2000년까지 미국이 자국 기술보호 등을 이유로 해외로 수출되는 암호 기술의 보안 수준을 40비트로 제한하자 1999년 한국 정보보호진흥원(현재의 한국인터넷진흥원)에서 국내 보안수준 향상을 위해 개발한 암호화 알고리즘의 이름을 쓰시오. (5점)

답 :

문제 19 기억장치에 관한 내용 중 국부성(Locality)의 개념을 설명하시오. (5점)

답 :

문제 20 스케줄링 기법에 관한 내용 중 시스템에서 특정 프로세스의 우선순위가 낮아 무한정 기다리게 되는 경우, 한 번 양보하거나 기다린 시간에 비례하여 일정 시간이 지나면 우선순위를 한 단계씩 높여 가까운 시간 안에 자원을 할당받도록 하는 기법을 가리키는 용어를 쓰시오. (5점)

답 :



최종점검 모의고사 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]

오픈 소스(Open Source)

[문제 2]

자료 흐름도(DFD; Data Flow Diagram)

[문제 3]

① 수평 분할(Horizontal Partitioning) ② 수직 분할(Vertical Partitioning)

[문제 4]

1 2 5 10

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 코마를 넣어 1, 2, 5, 10으로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

해설

문제의 코드는 입력받은 수의 약수를 배열에 저장한 후 출력하는 프로그램이다. 10의 약수를 구한다고 가정했을 때, 10의 약수는 10을 1부터 10까지 수로 차례대로 나누어 나머지가 0이 되게 하는 수 1, 2, 5, 10이 약수가 된다.

```
#include <stdio.h>
main() {
    ① int a, b, k[10];
    ② scanf("%d", &a);
    ③ ④ b = aliquot(a, k);
    ⑤ for (int j = 0; j < b; j++)
    ⑥     printf("%d ", k[j]);
}
⑦ int aliquot(int a, int k[]) {
    ⑧ int cnt = 0;
    ⑨ for (int i = 1; i <= a; i++)
    ⑩     if (a % i == 0) {
        ⑪         k[cnt] = i;
        ⑫         cnt++;
    }
    ⑬ return cnt;
}
```

- ① 정수형 변수 a, b와 10개의 요소를 갖는 정수형 배열 k를 선언한다.
- ② 정수를 입력받아 a에 저장한다.
- ③ a의 값과 배열 k의 시작 주소를 인수로 aliquot 함수를 호출한 다음 돌려받은 값을 b에 저장한다.
- ④ 리턴 값이 정수인 aliquot 함수의 시작점이다. ⑤번에서 전달받은 값과 주소는 각각 a와 배열 k가 받는다.
- ⑥ 정수형 변수 cnt를 선언하고 0으로 초기화한다.
- ⑦ 반복 변수 i가 1에서 시작하여 1씩 증가하면서 a보다 작거나 같은 동안 ⑧~⑩번을 반복 수행한다.
- ⑧ a를 i로 나눈 나머지가 0이면 ⑨, ⑩번 문장을 수행하고, 아니면 반복문의 처음인 ⑥번으로 돌아간다.
- ⑪ k[cnt]에 i의 값을 저장한다.



- ⑨ 'cnt = cnt + 1;'과 동일하다. cnt의 값을 1씩 누적한다.
 ⑩ cnt의 값 4를 호출한 곳으로 반환한다.
 ⑪ aliquot 함수로부터 반환받은 값 4를 b에 저장한다.
 ⑫ 반복 변수 j가 0에서 시작하여 1씩 증가하면서 b의 값인 4보다 작은 동안 ⑬번 문장을 반복 수행한다.
 ⑬ k[j]의 값을 출력한다.
 ※ 반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

	k[0]	k[1]	k[2]	k[3]	k[4]	k[5]	k[6]	k[7]	k[8]	k[9]
k[]	1	2	5	10						

main 함수				aliquot 함수				출력
a	b	j	k[j]	a	i	cnt	k[cnt]	
10	4	0	1	10	1	0	1	1 2 5 10
		1	2		2	1	2	
		2	5		3	2	5	
		3	10		4	3	10	
		4			5	4		
					6			
					7			
					8			
					9			
					10			
					11			

[문제 5]

갱신 이상(Update Anomaly)

[문제 6]

데이터 마트(Data Mart)

[문제 7]

① 5 ② 4

[문제 8]

① Hub & Spoke ② Message Bus 또는 ESB 방식

[문제 9]

sw *= -1

※ 답안 작성 시 주의 사항

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

해설

문제의 코드는 5행 5열의 배열에 'ㄹ'자 형태로 1부터 25까지의 수를 저장하여 출력하는 프로그램이다. 바깥쪽 for문의 i는 행 위치를, 안쪽 while문의 j는 열 위치를 담당하며, 안쪽 while문이 수행될 때마다 j의 증가값 역할을 하는 sw의 값을 +1 → -1 → +1 → -1 → +1로 바꾸는 과정을 통해 <출력>과 같이 배열에 값을 저장할 수 있다.

```
#include <stdio.h>
main() {
    ① int a[5][5], j = 0, sw = 1, k = 1;
    ② for (int i = 0; i < 5; i++) {
        ③ while (j <= 4 && j >= 0) {
            ④ a[i][j] = k++;
            ⑤ j += sw;
        }
        ⑥ sw *= -1;
        ⑦ j += sw;
    }

    ⑧ for (int i = 0; i < 5; i++) {
        ⑨ for (int j = 0; j < 5; j++)
        ⑩ printf("%3d", a[i][j]);
        ⑪ printf("\n");
    }
}
```

- ① 5행 5열의 크기를 갖는 정수형 배열 a와 정수형 변수 j, sw, k를 선언하고 각각 0, 1, 1로 초기화한다.
 ② 반복 변수 i가 0에서 시작하여 1씩 증가하면서 5보다 작은 동안 ③~⑦번을 반복 수행한다.
 ③ j가 4보다 작거나 같고 0보다 크거나 같은 동안 ④, ⑤번을 반복 수행한다.
 ④ a[i][j]에 k의 값을 저장한다. k는 후치 증가 연산이므로 연산을 마치면 1이 증가한다.
 ⑤ 'j = j + sw'와 같다. j에 sw의 값을 누적한다.
 ⑥ 'sw = sw * -1'과 같다. sw의 값에 -1을 곱한 값을 sw에 저장한다.
 ⑦ 'j = j + sw'와 같다. j에 sw의 값을 누적한다.
 ※ ②~⑦번 반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

i	j	sw	k	배열 a																									
	0	1	1																										
0	1	-1	2	<table border="1"> <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> </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 border="1"> <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> </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	-1	12	<table border="1"> <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																									
	2		13																										
	3		14																										
	4		15																										
	5		16																										
	4																												

3	3 2 1 0 -1 0	1	17 18 19 20 21	<table border="1"> <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 border="1"> <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																													

- ㉓ 반복 변수 i 가 0에서 시작하여 1씩 증가하면서 5보다 작은 동안 ㉓~㉔번을 반복 수행한다.
- ㉔ 반복 변수 j 가 0에서 시작하여 1씩 증가하면서 5보다 작은 동안 ㉔번을 반복 수행한다.
- ㉕ 3칸을 확보하여 $a[i][j]$ 의 값을 출력한다.
- ㉖ 다음 행을 출력하기 위해 커서를 다음 줄의 처음으로 옮긴다.

[문제 10]

동적 SQL(Dynamic SQL)

[문제 11]

- ① Quartz ② Cron

[문제 12]

스토리보드(Story Board)

[문제 13]

1 2 4 7

※ 답안 작성 시 주의 사항

프로그램의 실행 결과는 부분 점수가 없으므로 정확하게 작성해야 합니다. 예를 들어 출력값 사이에 콤마를 넣어 1, 2, 4, 7로 썼을 경우 부분 점수 없이 완전히 틀린 것으로 간주됩니다.

해설

문제의 코드는 선택 정렬 알고리즘을 이용하여 배열의 값들을 오름차순으로 정렬한 후 출력하는 프로그램이다. 선택 정렬은 첫 번째 자료를 두 번째 자료부터 마지막 자료까지 차례대로 비교하여 가장 작은 값을 찾아 첫 번째에 놓고, 두 번째 자료를 세 번째 자료부터 마지막 자료까지 차례대로 비교하여 그 중 가장 작은 값을 찾아 두 번째 위치에 놓는 과정을 반복하며 정렬을 수행한다. 1회전을 수행하고 나면 가장 작은 값의 자료가 맨 앞에 오게 되므로 그 다음 회전에서는 두 번째 자료를 가지고 비교한다. 마찬가지로 3회전에서는 세 번째 자료를 정렬한다.

```
public class Test {
    ㉓ static void arr(int[] a) {
    ㉔     int sw, temp, n = a.length;
    ㉕     for(int i = 0; i < n - 1; i++) {
    ㉖         sw = i;
    ㉗         for(int j = i + 1; j < n; j++)
    ㉘             if(a[j] < a[sw])
    ㉙                 sw = j;
```



```

10      temp = a[i];
11      a[i] = a[sw];
12      a[sw] = temp;
    }
  }
  public static void main(String[] args) {
1   int n[] = { 4, 2, 7, 1 };
2   arr(n);
3   for(int i:n)
4     System.out.printf("%d ", i);
  }
}

```

모든 Java 프로그램은 main() 메소드부터 시작해야 한다.

- ① 4개의 요소를 갖는 정수형 배열 n을 선언하고 초기화한다.
- ② 배열 n의 시작 주소를 인수로 하여 arr() 메소드를 호출한다.
- ③ 리턴 값이 없는 arr() 메소드의 시작점이다. ②번에서 전달받은 주소를 배열 a가 받는다.
- ④ 정수형 변수 sw, temp, n을 선언하고, n의 값을 배열 a의 길이인 4로 초기화한다.
 - length : 배열 클래스의 속성으로 배열 요소의 개수가 저장되어 있다. a 배열은 4개의 요소를 가지므로 a.length는 4를 가지고 있다.
- ⑤ 반복 변수 i가 0에서 시작하여 1씩 증가하면서 n-1보다 작은 동안 ⑥~⑫번을 반복 수행한다.
- ⑥ sw에 i의 값을 저장한다.
- ⑦ 반복 변수 j가 i+1에서 시작하여 1씩 증가하면서 n보다 작은 동안 ⑧, ⑨번을 반복 수행한다.
- ⑧ a[j]의 값이 a[sw]의 값보다 작으면 ⑩번을 실행한다.
- ⑨ sw에 j의 값을 저장한다.
- ⑩~⑫ 임시 변수 temp를 사용하여 a[i]와 a[sw]의 값을 교환한다.
- ⑬ 배열 n의 요소 수만큼 ⑭번을 반복 수행하는 향상된 for문이다.
 - int i : 배열 n의 각 요소가 할당될 변수를 선언한다.
 - n : 배열의 이름을 적어준다. 배열이 4개의 요소를 가지므로 각 요소를 i에 할당하면서 ⑭번을 4회 수행한다.
- ⑭ i의 값을 공백 한 칸과 함께 출력한다.

※ 반복문 실행에 따른 변수들의 값의 변화는 다음과 같다.

main() 메소드		arr() 메소드									출력
i	n[]	a[]	n	i	j	sw	a[j]	a[sw]	a[i]	temp	
	4 2 7 1	4 2 7 1	4								
		4 2 7 1 1 4		0	1 2 3 4	0 1 3	2 7 1	4 2 2 4	1	4	
		1 2 7 4 2		1	2 3 4	1	7 4	2 2 2	2	2	
		1 2 7 4 4 7		2	3 4	2 3	4	7 7	4	7	
				3							
1 2 4 7	1 2 4 7										1 1 2 1 2 4 1 2 4 7

[문제 14]

① 검증 테스트(Verification Test) ② 확인 테스트(Validation Test)

[문제 15]

테스트 케이스(Test Case)

[문제 16]

① VIEW ② AS SELECT

※ 답안 작성 시 주의 사항

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

해설

```
CREATE VIEW 학생_v AS
SELECT 번호, 이름, 학과
FROM 학생;
```

생성한 뷰의 이름은 <학생_v>이다.
'번호', '이름', '학과' 속성을 가져온다.
<학생> 테이블에서 속성을 가져와 뷰를 생성한다.

[문제 17]

다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

티어드롭 공격은 분할된 패킷의 순서를 기록하는 Fragment Offset 값을 변경하여 수신 측에서 재조립 시 오류로 인한 과부하를 유도하는 공격 방법이다.

[문제 18]

SEED

[문제 19]

다음 중 밑줄이 표시된 내용은 반드시 포함되어야 합니다.

국부성은 프로세스가 실행되는 동안 주기억장치를 참조할 때 일부 페이지만 집중적으로 참조하는 성질을 말한다.

[문제 20]

에이징(Aging) 기법