2022학년도 중등학교교사 임용후보자 선정경쟁시험

정보 · 컴퓨터

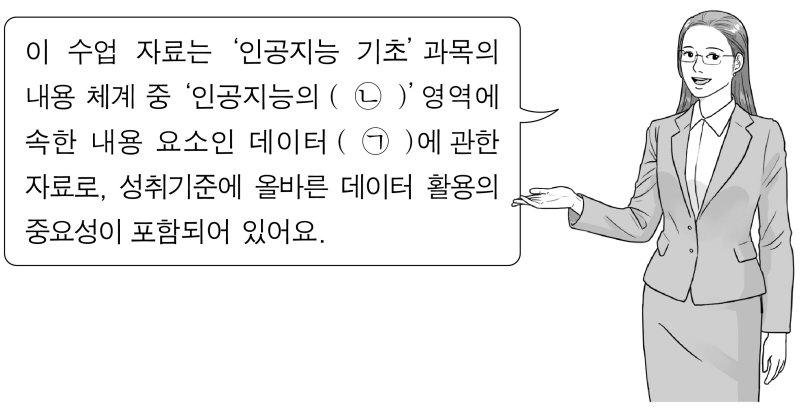
수험 번호 : (　　)　　**성　명 : (　　)**

제1차 시험	2교시 전공A	12문항 40점	시험 시간 90분

○ 문제지 전체 면수가 맞는지 확인하시오.
○ 모든 문항에는 배점이 표시되어 있습니다.

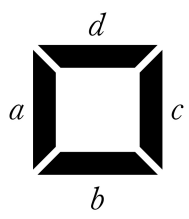
1. 다음은 정보 교사가 작성한 수업 자료 및 그에 대한 설명이다. 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 따른 '인공지능 기초' 과목에 제시된 괄호 안의 ㉠, ㉡에 해당하는 내용을 순서대로 쓰시오. [2점]

> 지난 10월 △△은 2014년부터 개발해 온 인공지능(AI) 채용 시스템을 폐기했다고 밝혔다. 프로그램을 테스트해 본 결과, 특정 성별을 상대적으로 높게 평가하는 현상이 나타나 자체 폐기한 것이다. 그동안 IT 기업 지원자 중에 특정 성별이 압도적으로 많았기 때문에, 이런 데이터를 학습한 AI가 문제를 일으킨 것이다. 또 다른 예로 재판에 사용하는 AI인 ◇◇는 특정 인종을 잠재적 범죄자로 잘못 분류할 확률이 2배 가량 높다는 사실이 밝혀졌는데, 이는 사람이 입력하는 데이터에 따라 (㉠) 이/가 생길 수 있음을 입증하는 결과이다.
>
> – ○○신문, 2018년 ○○월 ○○일자

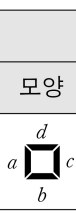
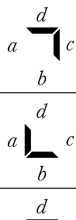
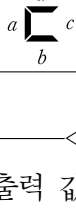
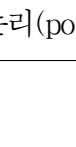


2. (가)는 4-세그먼트(segment) LED의 구성이고, (나)는 입력 X, Y에 따른 LED의 출력 모양과 LED a, b, c, d의 출력 값을 나타낸 것이다. <조건>을 고려하여 ㉠의 내용을 쓰고, 입력 X, Y에 따른 출력 a, b, c, d의 논리식을 각각 간소화하여 최소 SOP(minimum sum of products)로 쓰시오. [2점]

(가)



(나)

입력		출력				
X	Y	모양	a	b	c	d
0	0		1	1	1	1
0	1					
1	0					
1	1		1	1	0	1

㉠

<조 건>
○ LED a, b, c, d는 출력 값이 0이면 꺼지고 1이면 켜진다.
○ 회로의 동작은 정논리(positive logic) 방식을 따른다.

3. 다음 IPv4 라우팅 테이블(routing table)을 참조하여 목적지 주소가 140.17.7.7인 패킷에 해당하는 인터페이스(interface)와 201.80.1.3인 패킷에 해당하는 인터페이스를 순서대로 쓰시오. (단, 주소와 마스크는 CIDR(Classless Inter-Domain Routing) 표기법을 따름.) [2점]

mask	network address	next hop	interface
/24	201.80.44.0	–	a1
/23	201.80.2.0	–	b2
/16	201.80.0.0	–	c3
/16	140.17.0.0	–	d4
/16	140.27.0.0	–	e5
default	default	201.80.44.3	a1

4. (나)와 (다)는 (가)에 정의된 변수를 공유하며 병행 수행되는 C 유사 코드이다. <조건>을 고려하여 (라)의 괄호 안의 ㉢, ㉣에 해당하는 값을 순서대로 쓰시오. [2점]

(가)

```
int share = 0, delta = 0;
semaphore s1 = 0, s2 = 0, t = 1;
```

(나)

```
wait(t);
delta = delta + 1;
signal(t);
㉠ signal(s1);
  wait(s2);
wait(t);
share = share + delta;
signal(t);
```

(다)

```
wait(t);
delta = delta + 1;
signal(t);
㉡ wait(s1);
  signal(s2);
wait(t);
share = share + delta;
signal(t);
```

(라)

(나)와 (다)를 병행 수행했을 때 공유 변수 share가 갖게 되는 최종 값은 (㉢)이고, ㉠ 코드와 ㉡ 코드를 모두 제거한 후 (나)와 (다)를 병행 수행했을 때 공유 변수 share의 최종 값은 (㉢) 또는 (㉣)이/가 될 수 있다.

<조 건>

- ○ (가)의 변수 share, delta, s1, s2, t는 (나)와 (다)에 의해서만 공유되며, (나)와 (다), 혹은 ㉠과 ㉡을 제거한 (나)와 (다)는 항상 이들에 대한 초기화 작업 이후 병행 수행된다.
- ○ s1, s2, t는 이진 세마포어(semaphore) 변수로, 생성 시 s1과 s2는 0으로, t는 1로 초기화된다.
- ○ 이진 세마포어 변수 s에 대해 적용 가능한 원자 연산(atomic operation) wait(s)와 signal(s)는 다음과 같이 동작한다.
  - wait(s) : 이진 세마포어 변수 s의 값이 1이면 그 값을 0으로 변경한 후 바로 리턴되고, s의 값이 0이면 다른 프로세스의 signal(s) 연산 호출에 의해 깨어날 때까지 s에 연계된 큐에서 대기한 후 리턴된다.
  - signal(s) : 이진 세마포어 변수 s에 연계된 큐에 대기 중인 프로세스가 존재할 경우 그중 첫 번째 프로세스를 깨워준 후 리턴되고, 그렇지 않을 경우 s의 값을 1로 설정하고 리턴된다.

5. (가)는 알고리즘 수업에 관한 지도 교사와 예비 교사의 대화이고, (나)와 (다)는 예비 교사가 작성한 자료이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

예비 교사 : 선생님, 내일 수업을 위해 수를 맞히는 게임 자료 (나)를 만들었어요.
지도 교사 : 네. 학생이 재미있어 하는 소재이면서 알고리즘 수업에도 활용할 수 있겠네요.
예비 교사 : 그런데 어떤 관점으로 알고리즘 수업에 활용하면 좋을까요?
지도 교사 : 중학교의 교육과정에 제시된 내용 요소에 알고리즘 이해, 알고리즘 표현이 있고, 고등학교의 알고리즘 관련 내용 요소는 중학교의 내용을 기반으로 한 알고리즘 설계, (㉠)이/가 있어요. (㉠) 수업에서 (나)를 제시하고, 두 알고리즘 적용 사례에서 학생 A의 질문 횟수를 확인하는 활동을 해 보세요. 그리고 (㉡) 관점에서 질문의 횟수가 더 적은 것이 무엇인지를 알아보는 활동으로 구성해 보세요. (다)는 이 수업의 평가 기준으로 사용하면 좋겠네요.

(나)

[순차탐색 사례]	[이진탐색 사례]
학생 A : 1입니까?	학생 A : 10보다 큽니까?
학생 B : 아니요.	학생 B : 네.
학생 A : 2입니까?	학생 A : 15보다 큽니까?
학생 B : 아니요.	학생 B : 네.
학생 A : 3입니까?	학생 A : 18보다 큽니까?
학생 B : 아니요.	학생 B : 네.
… (중략) …	학생 A : 19보다 큽니까?
학생 A : 19입니까?	학생 B : 네.
학생 B : 아니요.	학생 A : 20입니까?
학생 A : 20입니까?	학생 B : 맞습니다.
학생 B : 맞습니다.	

(다)

수준	평가 기준
상	㉢
중	알고리즘의 연산을 몇 번 수행하는지 계산할 수 있다.
하	알고리즘에서 연산의 의미를 말할 수 있다.

<작성 방법>

- ○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 제시된 괄호 안의 ㉠에 해당하는 용어를 쓸 것.
- ○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 근거하여 괄호 안의 ㉡에 해당하는 용어를 쓸 것.
- ○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)의 성취기준과 (가)의 대화에 근거하여 (다)의 ㉢을 서술할 것. (단, 학생 행동에 해당하는 동사 2가지를 포함하되, 중/하 수준의 평가 기준은 제외함.)

6. (가)는 정보과 교사들 간의 대화이고, (나)는 A 교사가 컴퓨팅 사고력을 기반으로 '데이터와 기계학습' 영역에 대한 인공지능 기초 수업을 재구성한 것이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

A 교사 : 인공지능 기초 수업을 컴퓨팅 사고력 중심으로 재구성하려면 어떻게 해야 할까요?

B 교사 : 인공지능 기초 과목의 교과 역량인 컴퓨팅 사고력 중 (㉠)은/는 문제의 복잡성을 제거하기 위해 사용할 수 있습니다. (㉡)은/는 (㉠)에서 사용하는 기법 중 하나입니다.

C 교사 : 자동화도 컴퓨팅 사고력 과정에서 중요한 요소이며, (㉠) 과정을 통해 도출된 문제 해결 모델을 프로그래밍을 통해 구현할 수 있습니다.

… (중략) …

A 교사 : 인공지능을 활용한 문제 해결의 과정에서 적용할 수 있는 기계학습 모델은 어떤 것이 있나요?

B 교사 : 인공지능 기초 과목의 내용 체계에서 기계학습 모델의 내용 요소로 (㉢)을/를 제시하고 있습니다.

A 교사 : ㉣ 기계학습 알고리즘에 관한 수업은 어떻게 설계해야 할까요?

C 교사 : (㉤)

(나)

○ 차시별 지도 계획

학습 목표		1. 문제 해결에 필요한 자료를 수집하고 알고리즘을 설계할 수 있다. 2. 기계학습의 모델을 활용하여 프로그래밍하고 문제를 해결할 수 있다.
컴퓨팅 사고력 과정	학습 과정	수업 내용
㉠	자료 수집 및 분석	데이터를 수집하고 분석하기
	데이터 전처리	문제 해결에 불필요한 속성을 제거하고 필요한 속성으로 정제하기
	알고리즘 설계	(㉢)의 개념을 활용하여 문제 해결의 과정을 작성하기
자동화	프로그래밍	프로그래밍 언어를 활용하여 (㉢)을/를 구현하기
	적용	다양한 분야에서 수집된 데이터를 적용하여 문제 해결하기

<작성 방법>

○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 제시된 괄호 안의 ㉠, ㉡, ㉢에 해당하는 내용을 순서대로 쓸 것.

○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 따른 '인공지능 기초' 과목의 '교수·학습 방법 및 유의 사항'에 근거하여 밑줄 친 ㉣에 대한 대답으로 괄호 안의 ㉤을 서술할 것.

7. (가)는 A 교사가 작성한 교육과정 재구성을 위한 자료의 일부이고, (나)는 이를 발표한 후 토의한 내용이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

영역	프로그래밍 / 컴퓨팅 시스템
성취기준	[9정04-05] 실생활 문제 해결을 위한 소프트웨어를 협력하여 설계, 개발, 비교·분석한다. [9정05-02] ㉠ 센서를 이용한 자료 처리 및 동작 제어 프로그램을 구현한다.
교육과정 재구성 사례 발표 자료	
… (중략) … ○ 실생활 문제와 관련된 프로젝트 수행 시 가급적 ㉡ 컴퓨팅 시스템 영역과 연계하여 지도하며, 학습자 수준에 적절한 교육용 프로그래밍 언어와 교구를 선택하기 ○ 학습자 간 유의미한 상호작용과 문제 해결을 위한 프로그램 개발 및 수정 등 협력적 프로젝트의 일련의 수행 과정을 종합적으로 평가하기 ○ 센서 보드의 동작 설계와 프로그램 작성 과정을 공유하는 활동을 통해 효율적인 프로그램을 개발하도록 지도하기	

(나)

… (중략) …

A 교사 : ㉢ 일반적인 교수 체제 설계 모형과는 달리 교육 목적과 평가 요소들을 교육과정 설계에서 먼저 수립하고 교육용 프로그래밍 언어와 교구를 활용한 수업 활동을 계획하려고 합니다. 평가를 계획할 때 평가 요소로는 무엇을 고려해야 할까요?

B 교사 : 실생활 문제 해결을 위한 프로젝트 수행 과정을 관찰하고 ⓐ 포트폴리오를 활용하여 협력적 문제 해결의 과정을 종합적으로 평가할 수 있습니다. 이를 위해 ⓑ 프로그래밍 언어의 문법 이해 및 기능 습득과 관련한 평가가 중심이 되어야 합니다.

C 교사 : 센서 기반 프로그램을 구현하기 위해 ⓒ 학습자 수준과 학습 환경에 따라 센서의 종류와 개수를 결정하도록 해야 합니다. ⓓ 동작 설계와 프로그램 작성 과정보다는 센서보드 장치의 구성에 중점을 두고 지도해야 합니다.

<작성 방법>

○ 2015 개정 정보과 교육과정(교육부 고시 제2020-236호)에 제시된 내용 체계에서 밑줄 친 ㉠의 성취기준에 해당하는 밑줄 친 ㉡ 영역의 핵심 개념을 쓸 것.

○ 밑줄 친 ㉢에 해당하는 교수·학습 설계 모형을 쓸 것.

○ 평가 요소를 고려할 때 밑줄 친 ⓐ~ⓓ 중에서 틀린 부분 2가지를 찾고, 그 이유를 각각 서술할 것.

8. (가)는 출판사와 도서 정보를 관리하기 위한 관계형 데이터베이스이고, (나)는 도서 릴레이션을 생성하는 SQL문이다. (다)와 (라)는 (가)의 릴레이션에서 검색을 위한 SQL문과 실행 결과이다. <조건>을 고려하여 <작성 방법>에 따라 서술하시오. [4점]

(가)

출판사 릴레이션(기본키 : 출판사코드)

출판사코드	출판사명	대표이름
A2	가출판사	김재혁
A4	나출판사	정인국
B8	다출판사	최예원
C1	라출판사	이로빈

도서 릴레이션(기본키 : 도서코드, 외래키 : 출판사코드)

도서코드	제목	저자	평점	출판사코드
9-21	프로그래밍 언어	홍용소	5	A4
2-44	데이터베이스	김교육	5	A2
2-78	정보보호와 보안	최학권	3	A2
2-49	메타버스	임기요	1	C1
5-53	AI와 데이터 과학	라군아	4	A4

(나)

```
CREATE TABLE 도서 (
  도서코드      CHAR(4) NOT NULL,
  제목          VARCHAR(200) NOT NULL,
  저자          VARCHAR(45) NOT NULL,
  평점          INT NOT NULL [ ㉠ ],
  출판사코드    CHAR(2) NOT NULL,
  PRIMARY KEY(도서코드),
  FOREIGN KEY(출판사코드) REFERENCES 출판사(출판사코드)
);
```

(다)

```
SELECT 제목, 도서코드, 평점
    FROM 도서
    WHERE 제목 [ ㉡ ] ;
```

(라)

[SQL문]

```
SELECT A.출판사명, [ ㉢ ] AS 평균평점
    FROM 출판사 AS A, 도서 AS B
    WHERE A.출판사코드 = B.출판사코드
    [ ㉣ ] ;
```

[실행 결과]

출판사명	평균평점
가출판사	4.0
나출판사	4.5

<조 건>

- ○ (가)의 릴레이션은 도메인 제약조건, 참조 무결성 제약조건, 키 제약조건을 만족한다.
- ○ 도서 릴레이션의 출판사코드 속성은 출판사 릴레이션의 기본키인 출판사코드를 참조하는 외래키이다.

<작성 방법>

- ○ 도서 릴레이션의 평점 속성에 1~5의 정수 값만 저장될 수 있도록 (나)의 ㉠에 들어갈 내용을 쓸 것.
- ○ 제목에 '데이터'가 포함된 도서를 검색하려고 할 때 (다)의 ㉡에 들어갈 내용을 쓸 것.
- ○ (라)의 [실행 결과]와 같이 출판사별 출판 도서의 평균평점이 4.0이상인 출판사의 출판사명과 평균평점이 출력되도록 ㉢과 ㉣에 들어갈 내용을 쓸 것.

9. 다음은 배열을 사용해 문자열을 처리하는 C 프로그램이다. <작성 방법>에 따라 서술하시오. [4점]

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

int check(char *);

int main(void) {
   int i, volume, count = 0;
   char *words[] = {"A", "good", "deed", "Luck",
                    "students", "level", "rapport",
                    "reviver", "successful", "PASS"};

 ㉠printf("%s", *(words+4));

 ㉡volume = sizeof(words) / sizeof(char*);
   for(i = 0; i < volume; i++)
      if(check(words[i]))
         count++;

 ㉢printf("%d", count);
}

int check(char *word) {
   int pt = 0;
   int len = strlen(word);

   while(pt < (int)(len / 2)) {
      if(word[pt] != word[len - pt - 1])
         return 0;
      pt++;
   }
   return 1;
}
```

<작성 방법>

- ㅇ 밑줄 친 ㉠의 출력 결과를 쓸 것.
- ㅇ 밑줄 친 ㉡이 수행될 때 volume 변수에 저장되는 값을 쓸 것.
- ㅇ 밑줄 친 ㉢의 출력 결과를 쓰고, 그 출력 결과가 의미하는 바를 서술할 것.

10. 다음 C 프로그램의 수행 결과를 <작성 방법>에 따라 서술하시오. [4점]

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

int *f1(int, int *);

int main(void) {
   int *t = NULL;
   t = f1(0, t);
   *t = *t + 20;
   t = f1(1, t);
   free(t);
}

int *f1(int tag, int *p) {
 ㉠int a = 0;
   int s = 0;
   if(tag == 0) {
     p = (int *)malloc(sizeof(int));
     *p = 0;
   }
   a = a + 2;
   s = s + a;
   printf("%d  ", s);
   *p = *p + 10;
   s = s + *p;
   printf("%d  ", s);
   return p;
}
```

<작성 방법>

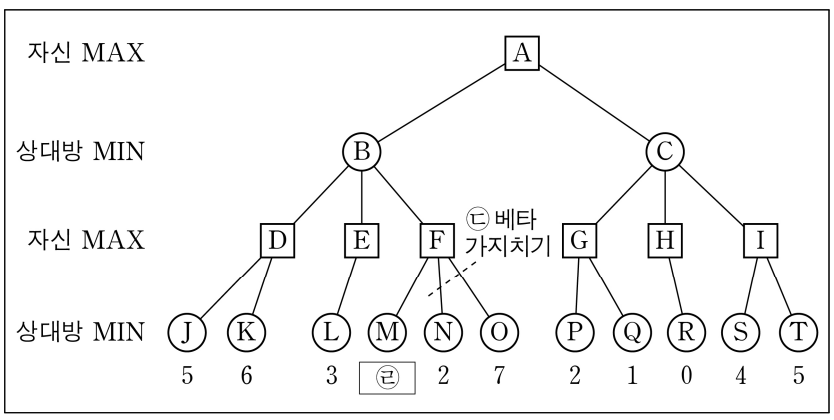
- ㅇ 위 프로그램을 수행할 경우 출력되는 값을 순서대로 쓸 것.
- ㅇ 밑줄 친 ㉠을 static int a = 0; 으로 변경하여 프로그램을 재수행할 경우 출력되는 값을 순서대로 쓸 것.

11. (가)는 mini-max 알고리즘을 개선한 알고리즘이고, (나)는 이 알고리즘을 적용할 게임 트리이다. <조건>을 고려하여 <작성 방법>에 따라 서술하시오. [4점]

(가)

```
alp_beta(node, pvalue) {
   if(node가 단말노드) return (node의 평가함수값);
   if(node가 최댓값노드) {
      value = -∞;
      for(node의 자식노드마다) {
         chvalue = alp_beta(자식노드, value);
         if(value < chvalue) value = chvalue;
         if(pvalue [ ㉠ ] value)
            break;   /* 가지치기 */
      }
      return value;
   } else {
      value = +∞;
      for(node의 자식노드마다) {
         chvalue = alp_beta(자식노드, value);
         if(value > chvalue) value = chvalue;
         if(pvalue [ ㉡ ] value)
            break;   /* 가지치기 */
      }
      return value;
   }
}
```

(나)



<조 건>

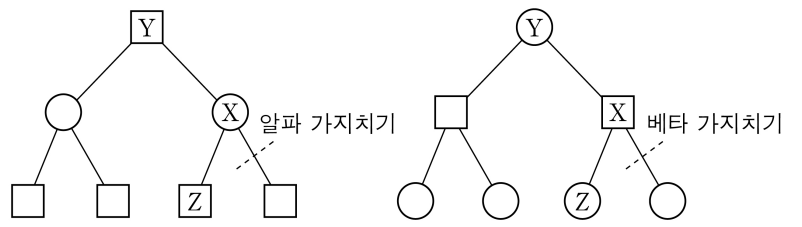
○ (나)는 두 선수가 번갈아 가며 플레이하는 게임을 진행할 때 일어날 수 있는 상황 전개 과정을 표현한 게임 트리이다. 게임 트리에서 단말노드 아래에 표시된 값은 해당 단말노드의 평가함수값(현재 플레이할 선수의 관점에서 게임 상황이 얼마나 유리한지 평가한 값)이고, mini-max 알고리즘은 이를 기초로 각 노드의 평가함수값을 계산한다.

○ mini-max 알고리즘은 게임 트리의 최댓값노드(자신)에서는 그 자식노드의 평가함수값 가운데 가장 큰 값을 선택하고, 최솟값노드(상대방)에서는 가장 작은 값을 선택한다. 각 노드의 평가함수값은 맨 좌측의 자식노드부터 시작해 각 자식노드에 대해 재귀적으로 구한 평가함수값을 이용해 계산된다.

○ (가)의 alp_beta 알고리즘은 mini-max 알고리즘에서 현재까지 계산된 평가함수값을 이용하여 아직 검사하지 않은 자식노드를 계산 대상에서 제외하는 가지치기 개념을 적용한 것이다.

○ alp_beta 알고리즘에서 최댓값노드(/최솟값노드)의 평가함수값을 구할 때 현 시점까지 찾아낸 가장 큰(/작은) 평가함수값을 그 노드의 알파(/베타) 값이라 한다.

○ 알파(/베타) 가지치기는 최솟값노드(/최댓값노드) X의 평가함수값을 구하는 과정에서 부모노드 Y의 알파(/베타) 값보다 작거나 같은(/크거나 같은) 평가함수값을 갖는 자식노드 Z가 발견될 경우 그 이후에 탐색하려 했던 X의 나머지 자식노드 모두를 가지치기하여 평가함수값 계산 대상에서 제외하는 것이다.



○ (나)의 단말노드 M의 평가함수값 ㉣은 ㉢ 베타 가지치기가 일어나도록 주어진다고 가정한다.

<작성 방법>

○ (가)의 alp_beta 알고리즘이 <조건>에 맞게 동작하도록 ㉠과 ㉡에 들어갈 내용을 순서대로 쓸 것.

○ (나)의 게임 트리에서 ㉢ 베타 가지치기가 일어나게 하는 ㉣의 값 중 최솟값을 쓸 것.

○ alp_beta 알고리즘을 수행할 경우 발생하는 가지치기 이름과, 가지치기로 인해 평가함수값 계산에서 제외되는 모든 노드의 이름을 쓸 것. (단, ㉢ 베타 가지치기는 제외함.)

12. (가)는 프로세스들이 병행 수행되고 있는 가상 메모리 시스템에서 시점 t의 프로세스 A의 적재 집합이고, (나)는 시점 t 이후 프로세스 A가 참조하게 될 페이지들의 번호를 참조 순서대로 나열한 것이다. <조건>을 고려하여 <작성 방법>에 따라 서술하시오. [4점]

(가)

[프로세스 A의 적재 집합]

페이지 번호	1	2	3	5	7	8	9
참조 비트	1	0	0	1	0	0	0

(나)

[시점 t 이후 프로세스 A의 페이지 참조열]

2 6 2 3 9 2 4 8 9 2 5 7 3

<조 건>

- ○ (가)는 시점 t에 프로세스 A의 적재 집합에 포함된 모든 페이지와, 해당 페이지 각각의 참조 비트(reference bit : 사용 비트(use bit)라고 불리기도 함.) 값을 보여 준다.
- ○ 프로세스 A의 시점 t 직전 페이지 참조에서 페이지 폴트가 발생하였다.
- ○ 프로세스 A는 시점 t 이후 (나)의 페이지들을 순서대로 한 번씩 참조하며 수행된다.
- ○ 추가 페이지 적재에 필요한 가용 프레임은 충분히 있다고 가정한다.
- ○ 시스템은 아래의 PFF(Page Fault Frequency) 정책을 적용해 프로세스별 적재 집합을 관리한다.
  - 프로세스가 페이지를 한 번씩 참조할 때마다 해당 프로세스의 가상 시간은 1씩 경과된다.
  - 페이지가 참조되면 그 참조 비트 값은 즉시 1로 설정된다.
  - 페이지 폴트 발생 시 해당 프로세스에 새로운 가용 프레임을 할당해 요구된 페이지가 적재되게 한 후, 직전 페이지 폴트 발생 시점부터 현 페이지 폴트 발생 시점까지 경과된 가상 시간이 4 이상일 경우, 해당 프로세스의 적재 집합에 속한 모든 페이지(새로 할당된 가용 프레임상의 페이지 제외)에 대해 참조 비트 값이 0인 페이지는 회수하고 참조 비트 값이 1인 페이지의 참조 비트 값은 0으로 설정한다.

<작성 방법>

- ○ 시점 t 이후 프로세스 A의 페이지 참조 중에 첫 페이지 폴트가 발생되는 페이지의 번호를 쓸 것.
- ○ 시점 t 이후 프로세스 A의 페이지 참조 중에 참조 비트 값이 0인 페이지가 회수되게 만드는 첫 페이지 폴트의 참조 대상 페이지 번호를 쓰고, 그때 회수되는 페이지들의 번호를 모두 나열할 것.
- ○ 프로세스 A가 (나)의 페이지 참조열 모두를 참조하게 될 때까지 발생하는 페이지 폴트의 총 횟수를 쓸 것.

<수고하셨습니다.>