# 2016학년도 중등학교교사 임용후보자 선정경쟁시험

# 정보 · 컴퓨터

**수험 번호 : (　　)**　　**성　명 : (　　)**

| 제1차 시험 | 2교시 전공A | 14문항 40점 | 시험 시간 90분 |
|---|---|---|---|

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

**1.** 다음은 중학교 정보 과목 수업을 준비하는 교사들의 대화와 교사들이 개발한 수업 자료이다. 2009 개정 교육과정에 따른 중학교 정보 과목 교육과정에 근거하여 괄호 안의 ㉠, ㉡에 해당하는 내용 체계의 영역명을 순서대로 쓰시오. [2점]

**<교사들의 대화>**

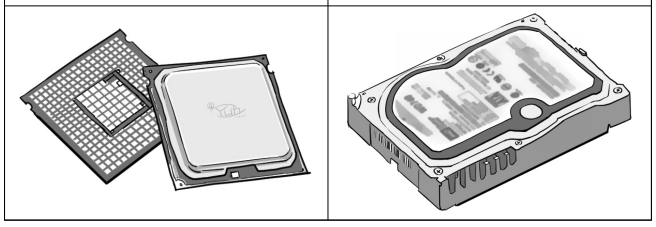
교사 A : 선생님, 제가 (　㉠　) 영역의 컴퓨터 구성 요소에 관한 수업에서 그림 자료를 사용했더니, 학생들의 흥미 유발 효과가 크지 않았습니다.

교사 B : 컴퓨터의 구성 요소에 관한 수업에서는 그림보다 실물을 보여 주는 것이 더 좋을 것 같습니다.

교사 A : 네, 그렇군요. 선생님의 수업 자료는 무엇에 관한 것인가요?

교사 B : 저는 (　㉡　) 영역의 정보 구조화에 관한 수업 자료를 만들었습니다.

**<교사 A의 수업 자료>**

| 수업 목표 | 컴퓨터 하드웨어의 주요 구성 요소를 구별할 수 있다. |
|---|---|
| 수업 자료 | 중앙처리장치 / 하드디스크  … (하략) … |

**<교사 B의 수업 자료>**

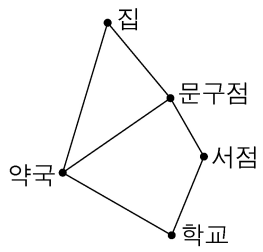
| 수업 목표 | 정보를 다양한 방법으로 구조화하여 표현할 수 있다. |
|---|---|

수업 자료

출석 통계

| 번호 | 이름 | 출석 일수 | 결석 일수 |
|---|---|---|---|
| 1 | 성춘향 | 21 | 0 |
| 2 | 이몽룡 | 21 | 0 |
| 3 | 변학도 | 20 | 1 |
| 4 | 홍길동 | 21 | 0 |
| 5 | 심청 | 21 | 0 |

학교 가는 길

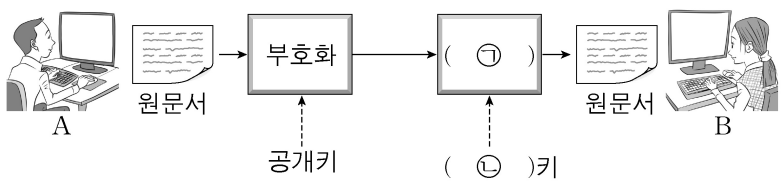



… (하략) …

**2.** 다음은 공개키 암호화 알고리즘을 이용하여 A가 B에게 문서를 보내는 과정을 나타낸 것이다. 괄호 안의 ㉠, ㉡에 해당하는 용어를 순서대로 쓰시오. [2점]

○ A가 보낸 문서는 부호화 과정과 (　㉠　) 과정을 거쳐 B에게 전달된다.
　• A는 공개키로 원문서를 부호화하여 전송한다.
　• B는 (　㉡　)키를 이용하여 전달받은 문서를 원문서로 변환하여 읽는다.

○ 공개키는 원문서를 부호화하기 위해 사용되고 원칙적으로 누구에게나 공개된다.




**3.** 다음은 정적 변수를 이용하는 C 언어 프로그램이다. 이 프로그램의 실행 결과를 쓰시오. [2점]

```
#include <stdio.h>
void sub(int n) {
        int j;
        static int s = 0;
        for(j = 1; j <= n; j++)
            s = s + j % 2;
        printf("%d ", s);
}
void main(void) {
        sub(5);
        sub(4);
}
```

**4.** 다음 <작업 내용>이 폭포수 모델의 소프트웨어 개발 생명주기(software development life-cycle)에서 어느 단계에 해당하는지 쓰시오. 그리고 이 단계에서 생산되는 산출물인 주요 형상관리 항목 중 1가지를 쓰시오. [2점]

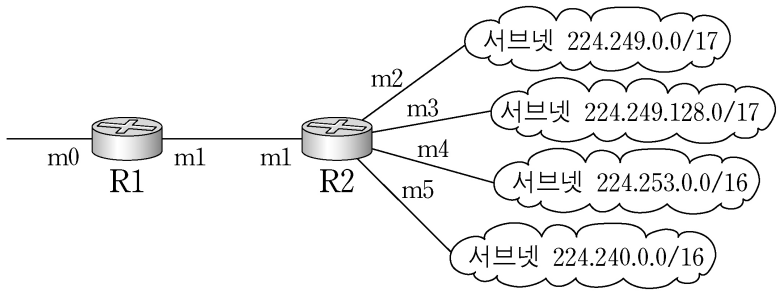
**<작업 내용>**

- ○ 소프트웨어 개발에서 무엇(what)을 개발할 것인가를 결정하는 단계이다.
- ○ 개발을 의뢰한 사용자의 요구나 주어진 문제를 정확히 파악하여 개발하려는 시스템에 대한 요구를 결정하는 작업을 수행한다.
- ○ 기존 시스템을 변경하는 경우 시스템의 현재 상태와 요구조건을 정의하고, 구현할 시스템의 목표를 도출한다.

**5.** (가)와 (나)는 <조건>에 따라 구성된 네트워크에 존재하는 라우터 R1과 R2의 라우팅 테이블(routing table)이다. (나)를 참조하여 (가)의 ㉠, ㉡에 해당하는 내용을 쓰시오. [2점]

<조 건>

- ○ 라우터 R1, R2와 출력 인터페이스 m0, m1, m2, m3, m4, m5에 해당하는 링크의 구성은 다음과 같다.




- ○ R1은 R2의 오른쪽에 있는 서브넷(subnet)으로 가는 패킷을 m1로, 그 이외의 패킷을 m0으로 전달한다.
- ○ 주소 표기는 IPv4에서 CIDR(Classless Inter-Domain Routing) 표기법을 따른다.

(가) R1 라우팅 테이블

| 목적지 서브넷 주소/마스크 | 출력 인터페이스 |
|---|---|
| ㉠ | m1 |
| ㉡ | m1 |
| 224.240.0.0/16 | m1 |
| default | m0 |

(나) R2 라우팅 테이블

| 목적지 서브넷 주소/마스크 | 출력 인터페이스 |
|---|---|
| 224.249.0.0/17 | m2 |
| 224.249.128.0/17 | m3 |
| 224.253.0.0/16 | m4 |
| 224.240.0.0/16 | m5 |
| default | m1 |

**6.** 단일 CPU에서 <조건>에 따라 명령어 A, B, C를 모두 처리하는 데 걸리는 최소 시간을 <작성 방법>에 따라 쓰시오. [2점]

<조 건>

○ 명령어 A, B, C를 처리하기 위해 필요한 시간은 다음 표와 같다.

| 명령어 | 단일 사이클 (single cycle) 구현 시 필요 시간 | 파이프라인 구현 시 구성 요소(component)별 필요 시간 | | | | |
|---|---|---|---|---|---|---|
| | | IF | ID | EX | MEM | WB |
| A | 120 | 40 | 20 | 40 | 20 | 0 |
| B | 140 | 40 | 20 | 40 | 0 | 40 |
| C | 160 | 40 | 20 | 40 | 20 | 40 |

○ 각 명령어는 한 번씩만 처리되며 처리 순서는 A, B, C이다.
○ 단일 사이클 구현에서 각 명령어는 한 클럭 주기(clock cycle)에 수행되며 클럭 주기 시간은 하나로 고정된다.
○ 파이프라인 구현에서 각 단계(stage)는 한 클럭 주기에 수행되며 클럭 주기 시간은 하나로 고정된다.
○ 파이프라인은 5단계(IF, ID, EX, MEM, WB)로 구성된다.
○ 파이프라인 구현에서 각 명령어는 5단계를 순서대로 모두 거친다.
○ 파이프라인에서 해저드(hazard)는 존재하지 않는다. 따라서 서로 다른 단계들은 충돌 없이 병렬로 수행될 수 있으며, 명령어 A, B, C는 서로 독립적이다.
○ 명령어 처리 시 위에서 제시한 시간 이외의 추가 소요 시간은 없다.

<작성 방법>

(1) 파이프라인을 쓰지 않고 단일 사이클로 구현할 경우, 명령어 A, B, C를 모두 처리하는 데 걸리는 최소 시간을 쓴다.
(2) 파이프라인으로 구현할 경우, 명령어 A, B, C를 모두 처리하는 데 걸리는 최소 시간을 쓴다.

**7.** 다음과 같은 두 개의 C 프로그램을 함께 실행하려면 외부 참조를 해결해야 한다. 외부 참조 문제는 번역 과정에서 생성된 변수와 함수에 대한 심벌 테이블(symbol table)을 이용하여 링킹 과정에서 해결한다. 번역 후 만들어진 심벌 테이블의 ㉠, ㉡에 해당하는 심벌을 순서대로 쓰시오. (단, 주소는 번역기가 별도의 과정을 통해 만든다고 가정한다.) [2점]

```
void func(void);
int aaa = 111;
float bbb = 3.14;
void main(void)
{
        func();
      … (중략) …
}
```

main.c

```
extern int aaa;
extern float bbb;
void func(void)
{
        bbb += aaa;
        … (중략) …
}
```

func.c

| 정의 | 외부참조 | 심벌 | 주소 |
|---|---|---|---|
| | √ | ㉠ | <address> |
| √ | | ㉡ | <address> |
| √ | | bbb | <address> |
| √ | | main | <address> |
| | | | |

main.c의 심벌 테이블

| 정의 | 외부참조 | 심벌 | 주소 |
|---|---|---|---|
| | √ | ㉡ | <address> |
| | √ | bbb | <address> |
| √ | | ㉠ | <address> |
| | | | |

func.c의 심벌 테이블

**8.** (나)는 (가)의 관계형 데이터베이스의 릴레이션 'employee'와 'department'에서 '부서별 평균 급여가 40000 이상인 부서의 부서번호, 부서명, 급여의 총액으로 구성된 뷰(view)를 생성'하는 SQL문이다. (나)의 ㉠, ㉡에 해당하는 내용을 순서대로 쓰시오. (단, 생성되는 뷰의 스키마는 ViewName1(employee.dno, department.dname, total)이다.) [2점]

(가)

employee (기본키 : eno)

| eno | ename | address | salary | dno |
|---|---|---|---|---|
| 12 | 손오공 | 서울 | 30000 | 5 |
| 33 | 이몽룡 | 서울 | 40000 | 5 |
| 98 | 변학도 | 광주 | 43000 | 1 |
| 77 | 심청 | 충북 | 25000 | 4 |
| 88 | 홍길동 | 부산 | 55000 | 1 |

department (기본키 : dno)

| dno | dname | mgr_eno |
|---|---|---|
| 5 | 연구부 | 33 |
| 4 | 행정부 | 77 |
| 1 | 인사부 | 88 |

○ 릴레이션 'employee'에서 eno는 사번, ename은 성명, address는 주소, salary는 급여, dno는 부서번호를 의미한다.
○ 릴레이션 'department'에서 dno는 부서번호, dname은 부서명, mgr_eno는 관리자 사번을 의미한다.
○ 릴레이션 'employee'의 외래키 employee.dno는 department.dno를 참조한다.

(나)

```
CREATE VIEW ViewName1 AS
SELECT employee.dno, department.dname, [ ㉠ ]
FROM employee, department
WHERE employee.dno = department.dno
GROUP BY employee.dno, department.dname
HAVING [ ㉡ ] ;
```

**9.** 다음은 중학교 정보 과목의 수업 자료 개발에 관한 두 교사의 대화이다. 대화 내용을 참조하여 밑줄 친 ㉠의 개념을 기술하고, 밑줄 친 ㉡이 저작권법(법률 제12137호, 2013.12.30., 일부개정)에 저촉되는지의 여부와 그 이유를 기술하시오. [4점]

교사 A : 선생님, 제가 직접 촬영한 사진과 동영상도 저작권의 보호를 받을 수 있나요?
교사 B : 네, 선생님의 사상이나 감정을 표현한 창작물이면 저작물로 저작권의 보호를 받을 수 있습니다.
교사 A : ㉠ 2차적저작물도 저작권의 보호를 받을 수 있나요?
교사 B : … (중략) …
교사 A : 그러면 ㉡ 인터넷에서 찾은 사진을 다운로드한 후, 편집하여 수업 자료로 사용하였다면 저작권법에 저촉되나요?

**10.** 다음은 중학교 정보 과목 수업을 설계하는 과정이다. 밑줄 친 ㉠의 개념을 기술하고, 밑줄 친 ㉡과 같은 교수·학습 방법을 선택한 이유를 '2009 개정 교육과정에 따른 중학교 및 고등학교 정보 과목 교육과정에 제시된 교육 내용의 연계성'에 근거하여 설명하시오. [4점]

| | |
|---|---|
| 1단계 | 중학교 정보 과목 교육과정 확인 |
| | 4. 내용의 영역과 기준<br>… (중략) …<br>③ 정렬과 탐색 방법의 이해<br>… (중략) …<br>· 실생활의 예를 통해 자료를 탐색하는 다양한 방법을 글과 그림을 통해 이해하고, 각 탐색 방법의 개념과 특성, 장단점을 설명한다.<br>… (중략) …<br>5. 교수·학습 방법<br>… (중략) …<br>다. 교수·학습 자료 및 활용<br>(1) 학습 동기를 유발하는 다양한 학습 자료를 활용하여 학습 성취도를 높이도록 한다.<br>(2) 다양한 매체와 ㉠ 언플러그드 학습 자료 등의 활용은 학습지도 계획에 반영되도록 한다.<br>… (하략) … |
| 2단계 | 수업 목표 설정과 학습 내용 선정 |
| | ○ 수업 목표 설정<br>순차 탐색과 이진 탐색의 개념과 특징을 설명할 수 있다.<br>○ 학습 내용 선정<br>순차 탐색과 이진 탐색의 개념과 특징 |
| 3단계 | 교과서 내용 확인과 교수·학습 방법 선택 |
| | ○ 교과서 내용 확인<br>순차·이진 탐색의 실생활 예와 탐색 방법의 개념과 특징<br>○ 교수·학습 방법 선택<br>㉡ 탐색의 원리와 방법에 따라 원하는 숫자 카드를 찾는 활동 |
| 4단계 | 교수·학습 지도안 작성과 수업 자료 제작 |
| | … (생략) … |

**11.** 다음은 피보나치 수를 이용한 탐색 알고리즘이다. 함수 $F_k$와 배열 A가 <조건>과 같을 경우 search(A, 6, 72)를 실행하였을 때 출력되는 값을 순서대로 쓰고, 밑줄 친 ㉠은 어떤 경우에 수행되는지 그 의미를 기술하시오. [4점]

**<알고리즘>**

```
/* A: 탐색 대상 키가 저장된 배열, A[i]: 배열 A의 i번째 값
   n: 피보나치 수의 색인
   find: 찾으려는 값 */
search(A, n, find) {
    k ← n;
    i ← Fk;
    while (k > 0) {
       print i;  /* i 값을 출력한다. */
       if (A[i] = find) return i;
       if (A[i] > find) {
          i ← i - Fk-2;
          k ← k - 1;
       } else {
          i ← i + Fk-2;
          k ← k - 2;
       }
    }
 ㉠ return -1;
}
```

<조 건>

○ 피보나치 함수: $F_k = F_{k-1} + F_{k-2}$ ($k \geq 3$, $F_1 = 1$, $F_2 = 1$)
( $F_1 = 1$, $F_2 = 1$, $F_3 = 2$, $F_4 = 3$, $F_5 = 5$, … )

○ 배열 A의 크기는 $F_{n+1} - 1$ 이다.

○ 배열 A의 초깃값:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | 5 | 18 | 21 | 33 | 43 | 48 | 59 | 61 | 64 | 72 | 84 | 91 |

**12.** 다음은 단방향 연결리스트를 이용한 스택에서 삽입 연산인 push( ) 함수를 C 언어로 작성한 프로그램의 일부이다. 빈칸 안의 ㉠, ㉡에 들어갈 내용을 순서대로 쓰시오. (단, 삽입되는 값은 스택의 최상위에 위치하며 스택에서 최하위 노드의 next는 NULL이다.) [4점]

```
#define OK          0
#define NOK        -1

struct node {         /* Linked stack의 노드 구조 */
  int value;          /* 노드의 원소 값 */
  struct node *next; /* 다음 노드를 가리키는 포인터 */
};

struct node *top = NULL;
   /* Linked stack의 최상위 원소를 가리키는 포인터 */

int push(int ikey)  /* ikey는 스택에 삽입되는 값 */
{
  struct node *ptr;

  ptr = (struct node *)
                  malloc(sizeof(struct node));
  if (ptr == NULL) {
        /* 메모리 할당에 실패하면 에러 코드를 반환함 */
    printf("Memory allocation is failed. \n");
    return NOK;
  }

  ptr->value = ikey;

  if (top == NULL) { /* 스택이 empty인 경우 */
    ptr->next = NULL;
  }
  else {              /* 스택이 empty가 아닌 경우 */
    [      ㉠      ];
  }
  [      ㉡      ];
  return OK;
}
```

**13.** 다음은 C 언어로 작성한 프로그램의 일부이다. func2( )는 func1( )을 재귀(recursive) 함수로 재작성한 것이다. (나)의 ㉠, ㉡에 해당하는 내용을 순서대로 쓰시오. (단, func1( ), func2( )로 전달되는 매개변수 k는 양수이다.) [4점]

(가)

```
int func1(int k)
{
  int i, r = 0;
  for(i = k; i > 0; i--)
    r += i;
  return r;
}
```

(나)

```
int func2(int k)
{
  if (k <= 1)
    return [  ㉠  ];
  else
    return [  ㉡  ];
}
```



한국교육과정평가원

**14.** <조건>과 같은 페이지 교체 정책을 사용하는 시스템에서 프로세스 A가 <참조 주소 열>과 같은 순서로 데이터를 참조할 때, 페이지 교체 과정을 <작성 방법>에 따라 기술하시오. [4점]

**<참조 주소 열>**

212, 36, 48, 256, 128, 365, 24, 400

<조 건>

- ○ 주소는 0부터 시작한다.
- ○ 페이지의 크기는 100이며, 페이지 번호는 0부터 시작한다. 예를 들어, 참조 주소 615의 페이지 번호는 6이다.
- ○ 프로세스 A에는 3개의 페이지 프레임이 고정적으로 할당되며, 초기 페이지 프레임은 모두 비어 있다.
- ○ 페이지 교체는 LRU(Least Recently Used) 알고리즘을 따른다.

<작성 방법>

(1) <참조 주소 열>에서 페이지 부재가 일어나는 주소를 순서대로 쓴다.

(2) 페이지 프레임에 최종적으로 남아 있는 페이지 번호를 모두 쓴다.

**<수고하셨습니다.>**