

2024학년도 중등학교교사 임용후보자 선정경쟁시험

정보 · 컴퓨터

수험 번호 : () 성 명 : ()

제1차 시험	2 교시 전공 A	12문항 40점	시험 시간 90분
--------	-----------	----------	-----------

- 문제지 전체 면수가 맞는지 확인하십시오.
- 모든 문항에는 배점이 표시되어 있습니다.

1. 다음은 2022 개정 정보과 교육과정의 교과 역량에 대해 정보 교사들이 나눈 대화의 일부분이다. 괄호 안의 ㉠과 ㉡에 해당하는 용어를 순서대로 쓰시오. [2점]

A 교사: 선생님, 2022 개정 정보과 교육과정을 도입하려고 보니 많은 내용이 개정된 것 같아요. 특히, 2015 개정 교육과정과는 다르게 정보 교과의 역량을 컴퓨팅 사고력, (㉠) 소양, (㉡) 소양으로 설정한 것 같아요.

B 교사: 맞아요. 컴퓨팅 사고력과 (㉡) 소양은 교육과정 총론의 '지식정보처리', '창의적 사고' 역량과 연계됩니다. 그리고 (㉠) 소양과 (㉡) 소양은 총론의 '협력적 소통', '공동체 역량'과 직접 연계됩니다.

A 교사: 네, 그럼에도 컴퓨팅 기반의 문제 해결을 위해 문제를 이해·분석하여 해결책을 설계하고, 실생활 또는 다양한 학문 분야의 문제를 해결하는 능력인 컴퓨팅 사고력은 2015 개정 교육과정의 용어와 동일하게 사용하네요.

B 교사: (㉡) 윤리의식과 데이터에 대한 이해를 기반으로 (㉡)을/를 통해 문제를 해결할 수 있는 능력인 (㉡) 소양과, 디지털 사회의 구성원으로서의 윤리의식과 시민성을 갖추고 디지털 기술을 기반으로 의사소통하고 협업할 수 있는 능력인 (㉠) 소양은 새롭게 제시된 것 같습니다.

2. (가)는 학사 관리 시스템의 학과 정보가 저장된 테이블과 학생 정보가 저장된 테이블이다. <조건>을 고려하여 <작성 방법>에 따라 쓰시오. [2점]

(가)

학과명	연락처	사무실
컴퓨터교육과	001-629-0002	사법관(B32) 2203호
컴퓨터공학과	001-629-0003	공학관(E13) 2132호

학번	이름	학과
20233893	이○신	컴퓨터교육과
20201790	김○신	컴퓨터공학과
20211775	유○순	컴퓨터공학과
20226933	강○찬	컴퓨터교육과
...

<조 건>

- (가)의 테이블들은 다음 SQL문을 통해 생성되었다.

```
CREATE TABLE 학과 (
  학과명 VARCHAR(16), 연락처 VARCHAR(16),
  사무실 VARCHAR(32), PRIMARY KEY(학과명)
);
```

```
CREATE TABLE 학생 (
  학번 VARCHAR(16), 이름 VARCHAR(16),
  학과 VARCHAR(16), PRIMARY KEY(학번),
  FOREIGN KEY(학과) REFERENCES 학과(학과명)
);
```

<작성 방법>

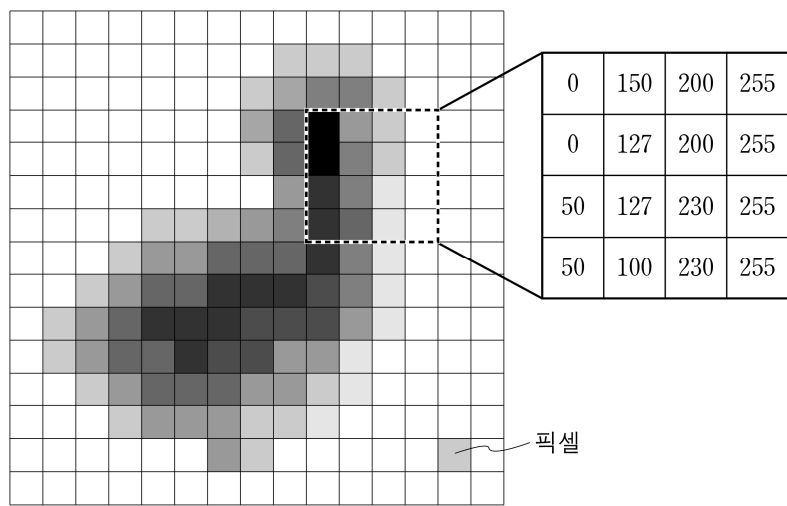
- 다음 SQL문 실행에 따라 위반되는 무결성 제약 조건을 쓸 것.

```
UPDATE 학생 SET 학과 = '인공지능학과' WHERE 학과 =
'컴퓨터공학과';
```

3. (가)는 그레이 스케일 이미지(gray scale image)에 대한 설명이고, (나)는 그레이 스케일 이미지 밝기를 조절하기 위한 C 프로그램의 일부이다. <조건>을 고려하여 <작성 방법>에 따라 쓰시오. [2점]

(가)

- 디지털 파일로 저장된 그레이 스케일 이미지를 구성하는 최소 단위를 픽셀(pixel)이라고 한다.
- 그레이 스케일 이미지의 각 픽셀에는 밝은 정도를 나타내는 숫자가 저장되며, 빨강, 녹색, 파랑 같은 컬러(color) 성분은 저장되지 않는다.
- 픽셀의 밝기는 0부터 255까지의 정수로 나타내며, 픽셀에 저장된 숫자를 픽셀값이라고 한다.
- 가장 작은 픽셀값 0은 검정색, 가장 큰 픽셀값 255는 흰색이고, 픽셀값이 작을수록 검정색에 가깝고 클수록 흰색에 가깝다.



(나)

```

...
unsigned char  origin[15][15];
unsigned char  bright[15][15];
unsigned char  whiteblack[15][15];
int           brightness, value, value2;
int           i, j;
...
printf("양의 정수를 입력하세요. : ");
scanf("%d", &brightness);

for (i = 0; i < 15; i++) {
    for (j = 0; j < 15; j++) {
        value = origin[i][j] + brightness;
        bright[i][j] =  ? 255 : value;
    }
}

for (i = 0; i < 15; i++) {
    for (j = 0; j < 15; j++) {
        value2 = bright[i][j]  7;
        whiteblack[i][j] = value2 ? 255 : value2;
    }
}

```

<조건>

- 원본 이미지와, 밝기가 조절된 결과 이미지에 대한 픽셀값은 unsigned char형의 2차원 배열에 저장된다.
- 밝기를 조절할 때, 그 값이 255를 넘어서는 경우에는, 해당 픽셀값을 최댓값인 255로 한다.
- 원본 이미지를 위한 픽셀값은 origin에 저장되어 있고, 밝게 조절된 픽셀값은 bright에 저장되며, 흰색과 검정색 2가지 픽셀 값만 갖도록 조절된 결과 이미지는 whiteblack에 저장된다.

<작성 방법>

- 원본 이미지의 픽셀값을 밝게 조절하기 위해 계산된 value의 값을 bright[i][j]에 저장하려고 할 때 ㉠에 들어갈 코드를 쓸 것.
- 밝게 조절된 이미지 픽셀값 bright[i][j]가 128보다 크거나 같으면 흰색으로, 128보다 작으면 검정색으로 만들기 위해 ㉡에 들어갈 비트 연산자를 쓸 것.

4. 다음에서 제시한 검사합(checksum) 계산 방법을 이용하여 2진수 메시지 100000 100000 000010에 대한 6비트 검사합을 구하고, 네트워크 프로토콜에서 검사합을 송신자가 수신자에게 전달하는 이유를 쓰시오. [2점]

1. 2진수 메시지를 6비트 단위로 분해한다.
2. 분해된 6비트들을 1의 보수 덧셈으로 모두 합한다. (6비트 2개에 대한 1의 보수 덧셈은 6비트들을 합한 결과가 $b_6b_5b_4b_3b_2b_1b_0$ 일 때, 캐리 비트 b_6 을 $b_5b_4b_3b_2b_1b_0$ 에 다시 더하는 것을 말한다. 예를 들어, 6비트들을 합한 결과가 1100000이면 캐리 비트 1을 100000에 다시 더하여 100001로 만든다.)
3. 2에서 나온 결과에 1의 보수를 취한다. (1의 보수를 취하는 것은 0을 1로, 1을 0으로 만드는 것을 말한다. 예를 들어, 100010에 대한 1의 보수는 011101이다.)

5. (가)는 2022 개정 중학교 정보과 교육과정에 따른 정보 과목 '내용 체계'의 일부이다. (나)와 (다)는 (가)를 근거로 작성한 교수·학습 지도안의 일부와 수업 활동지이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

핵심 아이디어	<ul style="list-style-type: none"> 알고리즘은 다양한 설계 전략을 통해 일상생활의 문제를 해결하는 데 활용된다. (㉠)을/를 고려해 설계된 알고리즘은 컴퓨터가 이해할 수 있는 언어로 구현되어 생활을 더욱 편리하게 하는 데 활용된다. 프로그램 개발은 협력이 필요하며, 공유하는 문화를 통해 더 좋은 프로그램이 개발된다.
구분 범주	내용 요소
가치·태도	<ul style="list-style-type: none"> 문제 분석을 통한 (㉡)의 중요성을 이해하고, 실생활 문제 해결을 실천하는 자세 문제 해결을 위한 다양한 해법을 탐색하고, 명확하게 알고리즘으로 표현하는 자세 프로그램의 효과성을 분석하고, 프로그램의 오류를 해결하려는 자세

(나)

교과 역량	㉢ 컴퓨팅 사고력	
성취기준	[9정03-04] 사례를 중심으로 문제 해결에 적합한 전략을 선택하여 알고리즘을 설계한다.	
교수·학습 단계	교수·학습 활동	
도입	동기 유발	<ul style="list-style-type: none"> 자판기 그림을 보여 주면서, 각 부분의 특징에 대해서 이야기해 본다. 학습 목표를 확인한다.
	문제 확인	<ul style="list-style-type: none"> (다)의 활동지에 나타난 음료수 자판기 문제 상황을 확인한다.
전개	해결 계획 설정	<ul style="list-style-type: none"> 문제를 해결하기 위한 계획을 설정한다. <ul style="list-style-type: none"> 문제에 관한 선행 지식(순차 구조, 선택 구조, 반복 구조)을 확인한다. 문제에 관한 학습 주제와 학습 계획을 설정한다.

(다)

수업 활동지

학습 문제: 다음은 음료수 자판기 알고리즘이다. 음료수 자판기가 제대로 작동되도록 알고리즘을 수정하시오.

```

graph TD
    Start([시작]) --> SetCount[음료수 개수 ← 5]
    SetCount --> Input[/돈 입력/]
    Input --> Is500{투입된 돈 = 500?}
    Is500 -- NO --> CalcChange[잔돈 ← 투입된 돈 - 500]
    CalcChange --> OutputChange[/잔돈 출력/]
    OutputChange --> OutputDrink[/음료수 출력/]
    Is500 -- YES --> OutputDrink
    OutputDrink -- a --> IsZero{음료수 개수 = 0?}
    IsZero -- YES --> StopSell[/판매 중지 출력/]
    StopSell --> End([종료])
    IsZero -- NO --> Input
  
```

[조건]

- 시작 시 음료수의 재고는 5개이다.
- 음료수의 가격은 500원이다.
- 돈 입력은 500원 이상만 가능하다.
- 재고가 없으면 음료수 자판기는 판매가 중지된다.
- 순서도는 다음 기호를 사용한다.

기호	명칭
○	단말 기호
□	처리 기호
◇	판단 기호
▱	자료 입출력
→	흐름선

<작성 방법>

- (나)의 밑줄 친 ㉢의 하위 요소 3가지에 근거하여, (가)의 괄호 안의 ㉠과 ㉡에 해당하는 용어를 쓸 것.
- (다)의 활동지의 알고리즘은 무한 반복이 발생하는 문제가 있다. 이 문제의 원인을 쓰고, 이를 해결하기 위해 ㉠ 위치에 들어갈 순서도 기호와 그 안에 들어갈 내용을 쓸 것.

6. (가)는 2022 개정 정보과 교육과정에 대한 정보 교사들의 대화 내용이고, (나)와 (다)는 2022 개정 정보과 교육과정에 따른 고등학교 선택 과목 중 일부 과목의 목표를 제시한 것이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

A 교사(중 학교): 선생님, 2022 개정 교육과정에 따라서 수업을 준비하고 있어요. 이번에 많은 변화가 있어서 한번 점검을 하고 수업을 진행하고 싶어요. 고등학교도 많은 변화가 있나요?

B 교사(고등학교): 네, 고등학교 정보 교과는 정보 과목 이외에 다양한 선택 과목을 운영할 수 있게 되었어요. 중학교는 어떤가요?

A 교사(중 학교): 중학교의 정보 교과는 기존 34시간 이상 편성·운영하는 기준에서 ㉠ 68시간 이상 편성·운영하는 것으로 개정이 되었어요.

B 교사(고등학교): 중학교 정보 교과의 시수가 많이 증가되었네요. 잘 고려해서 수업을 준비해야겠습니다.

(나)

(㉠) 과목 목표

(㉠)은/는 디지털 사회에서 실생활 및 다양한 학문 분야의 문제 해결에 융합되어 새로운 가치를 창출하는 소프트웨어의 가치와 필요성을 인식하고, 직면한 문제를 융합의 관점에서 효과적이고 창의적으로 해결하는 능력을 함양하여 사회에 기여할 수 있는 가치관과 태도를 기르는 데 중점을 둔다.

(다)

(㉡) 과목 목표

(㉡)은/는 컴퓨팅 사고력을 기반으로 디지털 사회에서 데이터의 역할 및 잠재적 가치와 데이터 과학에 기반한 문제 해결 과정의 중요성을 인식하며, 다양한 분야의 문제를 해결하고 합리적 의사 결정을 위한 통찰의 역량을 키우는 데 중점을 둔다.

<작성 방법>

- 중학교 정보 교과를 '㉠ 68시간 이상 편성·운영'하기 위한 방안을, 2022 개정 교육과정 총론의 '시간 배당 기준'에 제시된 용어를 활용하여 서술할 것.
- (나)의 괄호 안의 ㉠과 (다)의 괄호 안의 ㉡에 해당하는 정보 교과의 선택 과목명을 순서대로 쓸 것.

7. (가)는 2022 개정 정보과 교육과정에 따른 정보 과목의 '내용 체계' 일부를 학교급 구분 없이 나열한 것이다. (나)는 해당 과목의 '교수·학습 방법'을 참고하여 수업을 준비하는 교사 간의 대화이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

범주	구분	내용 요소
지식·이해	㉠	디지털 데이터 표현 방법
	㉡	디지털 데이터 압축의 효율성을 분석하고 평가하기
과정·기능	㉢	디지털 데이터 압축과 암호화
	㉣	효율적인 데이터 표현의 긍정적 측면을 활용하려는 자세
가치·태도	㉤	다양한 데이터를 디지털 데이터로 표현하기
	㉥	실생활의 많은 데이터가 디지털 형태로 변환되어 활용되는 긍정적 측면의 인식

(나)

신규 교사: 저는 디지털 데이터와 관련된 수업으로 정보 교육 과정의 교수·학습 방법을 참고해 문제기반학습(problem based learning) 모형을 적용할 예정입니다. 이를 위해 ㉠ 구조화된 문제를 개발하고, ㉡ 개발된 문제를 학습자 흥미를 고려해 제시하려고 합니다. 그리고 ㉢ 학생들에게 문제 해결 계획을 수립하기 위한 시간을 주고, ㉣ 모듈별로 협력해서 수행할 수 있도록 수업을 설계하려고 합니다.

수석 교사: 네, 그런데 문제기반학습으로 진행하려면 말씀하신 내용을 수정해야 할 필요가 있어 보입니다.

<작성 방법>

- 2022 개정 중학교 정보과 교육과정의 '내용 체계'에 따라 수업을 계획할 때 (가)의 ㉠~㉥ 중에서 지식·이해에 해당하는 기호 1가지와 과정·기능에 해당하는 기호 1가지를 순서대로 쓸 것.
- (나)의 신규 교사가 제시한 밑줄 친 ㉠~㉣ 중 문제기반학습에 따른 특징으로 적절하지 않은 것을 1가지 찾고, 그 이유를 서술할 것.

8. (가)는 로봇 걸음 문제와 해결 방법에 대한 설명이고, (나)는 (가)의 해결 방법을 구현한 robot 함수를 포함한 C 프로그램이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

- 로봇 걸음 문제는 한 걸음에 1미터 또는 2미터 또는 3미터를 이동할 수 있는 로봇이 있다고 가정했을 때, 이 로봇이 n 미터를 이동할 수 있는 방법의 수, $robot(n)$ 을 구하는 문제이다.
- 작은 수에서부터 예를 들어보면, 1미터를 이동하는 방법은 1가지만 있으므로, $robot(1) = 1$ 이다.
- 2미터를 이동하는 방법은 1미터로 2번 이동하는 방법, 또는 2미터로 1번 이동하는 방법이 있다. 따라서 $robot(2) = 2$ 이다.
- 3미터를 이동하는 방법은 3가지로 나뉘어진다.
 - 1미터를 한 걸음에 이동한 후 남은 2미터를 이동하는 방법
 - 2미터를 한 걸음에 이동한 후 남은 1미터를 이동하는 방법
 - 3미터를 한 걸음에 이동하는 방법
 따라서 $robot(3) = robot(2) + robot(1) + 1 = 4$ 이다.
- $n > 3$ 인 경우, n 미터를 이동하는 방법은 3미터를 이동하기 위한 방법처럼 3가지로 나뉘어진다.
 - 1미터를 한 걸음에 이동한 후 남은 $(n-1)$ 미터를 이동하는 방법
 - 2미터를 한 걸음에 이동한 후 남은 $(n-2)$ 미터를 이동하는 방법
 - 3미터를 한 걸음에 이동한 후 남은 $(n-3)$ 미터를 이동하는 방법

(나)

```
#include <stdio.h>

int robot(int n) {
    if(n == 1 || n == 2) ㉠;
    if(n == 3) return 4;
    return ㉡;
}

int main() {
    int distance, walk;
    printf("로봇이 이동할 거리: ");
    scanf("%d", &distance);
    walk = robot(distance);
    printf("%d", walk);
    return 0;
}
```

<작성 방법>

- (나)의 robot 함수가 올바르게 동작하도록 ㉠과 ㉡에 들어갈 코드를 순서대로 쓸 것.
- 프로그램 실행 시 6을 입력하면 출력되는 결과를 쓸 것.
- 프로그램 실행 시 36을 입력한 경우와 37을 입력한 경우의 출력 결과가 다음과 같이 나왔다. 37을 입력한 경우의 출력 결과가 음수 값으로 나온 이유를 쓸 것.
 - 36을 입력한 경우의 출력 결과: 2082876103
 - 37을 입력한 경우의 출력 결과: -463960867

9. (가)는 2진수 $(-1.011)_2$ 을 IEEE 754 부동소수점 표준의 32비트 단일 정밀도(single-precision) 형식으로 표현한 것이다. <조건>을 고려하여 <작성 방법>에 따라 서술하시오. [4점]

(가)

부호 (sign)	지수 (exponent)	가수 (fraction)
1	0111 1111	011 0000 0000 0000 0000
1비트	8비트	23비트

<조건>

- IEEE 754 부동소수점 표준의 32비트 단일 정밀도 형식으로 표현된 수의 값은 $(-1)^{\text{부호}} \times 2^{(\text{지수}-127)} \times (1.\text{가수})$ 이다. (단, $0, \pm\infty, \text{NaN}$, 비정규화된 수는 예외로 한다.)
- 정규화된 2진수의 가장 앞쪽 1비트는 가수에서 생략하고 표현하지 않는다.

<작성 방법>

- 2진수 $(1.011)_2$ 을 IEEE 754 표준의 32비트 단일 정밀도 형식으로 표현할 것.
- 2진수 $(10.1)_2$ 을 IEEE 754 표준의 32비트 단일 정밀도 형식으로 표현할 것.
- IEEE 754 표준의 32비트 단일 정밀도 형식으로 표현된 2진수 $(1.011)_2$ 과 $(10.1)_2$ 을 '2진 부동소수점 곱셈'한 결과를 IEEE 754 표준의 32비트 단일 정밀도 형식으로 표현하고, 이를 구하는 과정을 서술할 것.

10. (가)는 그레이 코드(Gray code)의 특징이고, (나)는 그레이 코드를 위한 변환 규칙이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

- 각 비트의 위치에 가중치가 부여되는 산술용 코드가 아니다.
- 서로 인접한 코드 간에 하나의 비트만 다르다. 예를 들어, 다음 코드는 서로 인접한 코드 간에 하나의 비트만 다르다.

2진수	그레이 코드	
...	...	
0 0 1 0	0 0 1 1	} 인접한 코드
0 0 1 1	0 0 1 0	
0 1 0 0	0 1 1 0	} 인접한 코드
...	...	

(나)

n 비트 2진수 $b_{n-1} \dots b_1 b_0$ 과 n 비트 그레이 코드 $g_{n-1} \dots g_1 g_0$ 간의 변환 과정은 다음과 같다.

- 2진수 → 그레이 코드 변환
그레이 코드의 최상위 비트 g_{n-1} 은 2진수의 최상위 비트 b_{n-1} 과 같다. 2진수 비트의 왼쪽에서 오른쪽으로 진행하면서 인접한 두 개의 비트를 더한 합이 그레이 코드 비트 g_i 가 된다. 더할 때 발생하는 캐리는 무시한다.

$$g_i = b_{i+1} + b_i \quad (i = n-2, \dots, 1, 0)$$

- 그레이 코드 → 2진수 변환
2진수의 최상위 비트 b_{n-1} 은 그레이 코드의 최상위 비트 g_{n-1} 과 같다. 그레이 코드 비트의 왼쪽에서 오른쪽으로 진행하면서 그레이 코드 비트 g_i 와 직전에 생성된 2진수 비트 b_{i+1} 을 더한 합이 2진수 비트 b_i 가 된다. 더할 때 발생하는 캐리는 무시한다.

$$b_i = g_i + b_{i+1} \quad (i = n-2, \dots, 1, 0)$$

—<작성 방법>—

- 6비트 2진수 $b_5 b_4 b_3 b_2 b_1 b_0$ 이 110011일 때, 이 2진수를 (나)의 변환 규칙에 따라 그레이 코드로 변환한 결과를 쓸 것.
- 6비트 그레이 코드 $g_5 g_4 g_3 g_2 g_1 g_0$ 이 100101일 때, 이 그레이 코드를 (나)의 변환 규칙에 따라 2진수로 변환한 결과를 쓸 것.
- (나)의 2진수 → 그레이 코드, 그레이 코드 → 2진수 변환 과정에서 덧셈 연산을 논리 게이트로 대체할 때, 공통적으로 사용할 수 있는 논리 게이트를 쓰고, 그 이유를 서술할 것. (단, 코드 1비트당 1개의 논리 게이트를 사용한다.)

11. (가)는 기수 정렬을 수행하는 C 프로그램의 일부이고, (나)는 (가)에서 사용된 함수에 대한 설명이다. <작성 방법>에 따라 서술하시오. [4점]

(가)

```
#define BUCKETS 10
#define DIGITS 5

void radix_sort(int list[], int cnt) {
    int b, d, i, factor = 1;
    for(b = 0; [㉠] ; b++) init_queue(b);
    for(d = 0; d < DIGITS ; d++) {
        for(i = 0; [㉡] ; i++)
            enqueue( (list[i] / factor) % 10, list[i]);

        for(b = i = 0; b < BUCKETS; b++)
            while(!is_empty(b))
                list[i++] = dequeue(b);

        for(i = 0; i < cnt; i++) printf("%d ", list[i]);
        printf("\n");

        [㉢] factor *= 10;
    }
}
```

(나)

- radix_sort(배열 주소, 배열 크기) 함수는 5자리 이내의 10진 정수들이 저장된 배열의 주소와 그 배열의 크기를 입력 받아, 전역변수로 선언된 10개의 큐를 이용하여 오름차순으로 정렬하는 함수로서, 기수 정렬 알고리즘을 사용한다.
- 각각의 큐는 고유한 QID(0~9)를 가진다.
- init_queue(QID)는 QID에 해당하는 큐를 초기화하는 함수이다.
- is_empty(QID)는 QID에 해당하는 큐가 공백인 경우에 1을 반환하고, 아닌 경우에 0을 반환하는 함수이다.
- enqueue(QID, 데이터)는 QID에 해당하는 큐에 데이터를 인큐(enqueue, 삽입)하는 함수이다.
- dequeue(QID)는 QID에 해당하는 큐에서 데이터를 디큐(dequeue, 삭제)하여 반환하는 함수이다.

—<작성 방법>—

- radix_sort 함수가 올바르게 동작하도록 ㉠과 ㉡에 들어갈 코드를 순서대로 쓸 것.
- 정렬 과정을 수행할 때, 밑줄 친 ㉢에서 곱셈 연산을 하는 이유를 쓸 것.
- 배열 a의 값이 {86, 48, 25, 45, 67, 35, 22, 82}인 상태에서 radix_sort(a, 8)을 호출하는 경우, 출력 결과의 첫 줄을 쓸 것.

12. 다음은 단일 CPU 시스템 기반의 멀티레벨 피드백 큐(multilevel feedback queue)에 저장된 프로세스들의 상태이다. <조건>을 고려하여 <작성 방법>에 따라 서술하시오. [4점]



<조 건>

- 0 레벨 큐의 우선 순위가 가장 높으며, 2 레벨 큐의 우선 순위가 가장 낮다.
- 높은 우선 순위 큐가 비워진 후, 낮은 우선 순위 큐에 저장된 프로세스가 실행된다.
- 최하위 우선 순위 큐를 제외하고, 프로세스는 자신이 위치한 큐의 CPU 할당 시간을 모두 사용한 경우, 한 레벨 낮은 우선 순위 큐로 이동하여 잔여 작업을 실행한다.
- 2 레벨 큐는 라운드 로빈(RR, Round Robin) 방식으로 프로세스를 선택하며, 선택한 프로세스별로 20ms(millisecond)씩 CPU 자원을 할당한다.
- 1 레벨 큐와 0 레벨 큐는 선입 선처리(FCFS, First Come First Served) 방식으로 프로세스를 선택하며, 선택한 프로세스별로 20ms씩 CPU 자원을 할당한다.
- 현재 상태에서 추가로 프로세스가 멀티레벨 피드백 큐에 진입하지 않는다.
- 프로세스 ID는 멀티레벨 피드백 큐에 진입한 순서대로 부여된 고유한 값이다.
- 각 프로세스는 입력출력요청(I/O burst)이 발생하지 않는 프로세스 작업으로만 구성된다.
- CPU 자원을 할당하고 회수하는 프로세스 간 문맥교환(context switching) 시간은 0ms로 가정한다.
- CPU 할당 시간이 종료되기 전에, 프로세스가 작업을 종료할 경우 운영체제는 즉시 해당 자원을 회수한다.
- 프로세스에 CPU 자원이 할당되어 작업을 수행하는 동안 인터럽트(interrupt) 등으로 인한 강제 자원 회수는 발생하지 않는다.

<작성 방법>

- 현재 상태에서 45ms 후 1 레벨 큐에 저장된 모든 프로세스의 ID를 쓸 것.
- 프로세스 P6이 작업을 종료할 때까지 소요되는 시간을 ms 단위로 쓸 것.
- 프로세스가 자신의 작업을 마치고 종료하는 순서대로 프로세스 ID를 쓸 것.

<수고하셨습니다.>