

## 프로그래밍언어론

### 2015년 시행 5급(기술) 공채 제2차시험

응시번호 :

성명 :

제 1 문. 이항연산자  $\Delta$ 와  $\nabla$ 를 가지는 수식에 대한 다음 문법을 보고 물음에 답하시오.  
(총 10점)

$\langle \text{expr} \rangle \rightarrow \langle \text{expr} \rangle \Delta \langle \text{expr} \rangle \mid$   
 $\langle \text{expr} \rangle \nabla \langle \text{expr} \rangle \mid$   
 $(\langle \text{expr} \rangle) \mid \langle \text{id} \rangle$   
 $\langle \text{id} \rangle \rightarrow a \mid b \mid c$

- 1) 이 문법이 모호함(ambiguous)을 보이시오. (5점)
- 2) 이항연산자  $\Delta$ 와  $\nabla$ 는 우결합(right-associative)이며  $\Delta$ 가  $\nabla$ 보다 우선순위가 높다고 가정하고, 결합규칙과 우선순위에 맞도록 모호하지 않은 문법으로 수정하시오. (5점)

제 2 문. 제네릭 프로그래밍(generic programming)이란 자료형(data type)에 관계없이 공통되게 동작하는 알고리즘에 대해서 자료형을 매개변수로 받아 사용할 수 있도록 지원하는 프로그래밍 기법이다. 최근 C++와 Java에서는 제네릭 프로그래밍을 지원함으로써 소프트웨어의 재사용성을 높이고 있다. 다음 물음에 답하시오.  
(총 15점)

- 1) 두 개의 값을 매개변수로 받아서 큰 값을 리턴하는 my\_max 함수를 제네릭 프로그래밍을 이용해서 작성하려고 한다. C++ 언어를 이용해서 작성하시오. (5점)
- 2) 두 개의 값을 매개변수로 받아서 큰 값을 리턴하는 myMax 함수를 제네릭 프로그래밍을 이용해서 작성하려고 한다. Java 언어를 이용해서 작성하시오. (5점)
- 3) 제네릭 프로그래밍을 다형성(polymorphism) 관점에서 객체지향 언어의 상속 다형성과 비교하여 설명하시오. (5점)

제 3 문. 다음과 같은 역할을 하는 map 연산과 reduce 연산이 있다고 하자.

- map 연산은 입력으로 주어진 각 항목에 대해서 동일한 연산을 적용하여 (key value) 쌍을 출력한다. 예를 들어, (map square (1 2 3))은 (1 2 3) 각각에 제곱을 구하여 ((1 1) (2 4) (3 9))와 같이 (key value) 형태로 출력한다.
- reduce 연산은 key와 해당 key의 모든 value들을 입력으로 받아, 그 value들에 대해 주어진 연산을 적용하여 하나의 (key value) 쌍을 출력한다. 예를 들면, (reduce sum (A (1 2 4)))는 A를 key로 갖는 value들이 (1 2 4)이며, 이 value들의 sum을 구하여 (key value) 형태인 (A 7)을 출력한다.

다음 물음에 답하시오.

(총 10점)

- 1) 주어진 문서 파일에 있는 단어들의 출현 빈도를 map과 reduce를 통해 계산할 수 있다. 이때 사용할 map과 reduce를 각각 pseudo code로 작성하시오. (5점)
- 2) 이러한 map과 reduce를 사용하여 분산처리환경에서 대규모 데이터를 처리할 경우 어떤 장점이 있는지 설명하시오. (5점)

제 4 문. 다음 C 프로그램에 대하여 물음에 답하시오.

(총 15점)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char *getString();
char *concat(char *a, char *b);
void main() {
    char *a, *b, *buffer;
    while (1) {
        a = getString();
        b = getString();
        buffer = concat(a,b);
        printf("%s + %s : %s\n", a, b, buffer);
    }
}

char *getString() {
    char buf[100];
    gets(buf);
    return buf;
}

char *concat(char *a, char *b) {
    char *c = malloc(sizeof(char)*(strlen(a)+strlen(b)+1));
    strcpy(c, a);
    strcat(c, b);
    return c;
}
```

- 1) 위 프로그램은 buffer overflow를 발생시킬 수 있다는 문제가 있다. 문제의 발생 지점, 원인과 해결방법을 설명하시오. (5점)
- 2) 위 프로그램은 dangling reference 문제가 발생한다. 문제의 원인과 해결방법을 설명하시오. (5점)
- 3) 위 프로그램을 계속 수행하게 되면 메모리 부족으로 프로그램이 비정상적으로 종료될 수 있다. 이러한 상황이 발생하는 이유와 해결방법을 설명하시오. (5점)

## 인사혁신처 시험출제과장