**Test #7. 2012.4.26 프로그래밍언어론 01 분반- 개인 문제**

학번:

이름:

조번호:

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
2. C에서 int 변수 a와 b의 값이 각각 0일 때, (a >= b) || (b=getchar()) 의 결과는 입력된 문자값을 가진다.
3. C에서 int 변수 a의 값이 1일 때, b = (++a)+a 의 값과 b = (a++)+a의 값은 같다.
4. ALGOL 60의 for index :=0 step 4 until 8은 다음 Fortran 86 프로그램과 동일하다.

DO i=0, 8, 4

stmt

END DO

1. goto의 목적지는 goto보다 더 안쪽으로 nested 된 statement 그룹 내부가 될 수 없다.
2. 다음 Java 의 능동 iterator는 hasNext()가 참일 때는 gradeList.next(); 는 null이 아니다.

List grades = new List();

Iterator gradeList = grades.iterator();

while ( gradeList.hasNext() ) {

listItem = gradeList.next();

print ( listItem );

}

1. C의 continue는 loop의 iteration횟수를 줄여준다.
2. Ada는 exit을 쓰면 하나의 레벨에서만 제어가 loop 밖으로 나가지만, C의 break는 어떠한 수의 레벨에서도 제어가 loop 밖으로 나갈 수 있다.
3. Fortran의 IF (산술식) N1, N2, N3 과 같은 표현에서 N1은 비교대상 값을 나타내고, N2, N3는 프로그램의 label을 나타낸다.
4. C, C++, Java의 switch 문에서 break를 생략해도 생략 전과 동일한 결과를 가진다.
5. 프로그래밍 언어에서 nested if 를 제공하지 않으면 dangling else 문제가 발생하지 않는다.

2. Smalltalk 에서 사용되는 수동 iterator의 예를 나타낸 코드이다. 다음 밑줄 그은 빈칸에 들어가는 내용은? (단, listElements.length()는 listElements가 가지는 원소의 개수를 나타낸다.)

class List {

Object[] listElements = new Object[size];

...

public void do( Function userOperation ) {

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

userOperation( listElements[k] );

}

}

List grades = new List();

aFunction = (item){ print(item) }; // 수행할 메소드

...

grades.do( aFunction ); // 메소드 전달

3. 연산자 오버로딩 단점을 적고 단점이 발생되는 예를 적으시오.

**Test #7. 2012.4.26 프로그래밍언어론 01 분반 조별 문제**

조번호:

참여자 (학번, 이름)

-

-

-

-

-

-

1. 연상
2. C의 int 타입 데이터를 가지는 linked-list에 대해 능동 iterator를 정의하고 사용 예를 보이시오. (hint: hasNext(), next() 와 같은 함수를 정의한 후, 필요한 데이터를 모아 가지고 있는 구조체 Iterator 를 정의하여 이 함수들의 인자로 사용. )