**Test #4. 2012.4.2 프로그래밍언어론 00 분반- 개인 문제**

학번:

이름:

조번호:

역할 : 사회자, 일반 조원 ........................ (자신의 팀 내의 역할에 동그라미 하시오.)

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
2. 어떤 Subprogram이나 block에 대한 nonlocal variable (비지역변수)는 해당 subprogram/block에 대해 가시적이지 않으면서 그 block에 선언되어 있지 않은 변수를 말한다.
3. C는 중첩된 subprogram을 정의할 수는 없지만 subprogram내에 block을 중첩시킬 수 있다.
4. static scope rule 에서 자신을 직접 둘러싸고 있는 subprogram 을 static parent 라고 한다.
5. C에서 전역 변수 x와 동일한 이름의 local 변수 x가 선언되었을 때 local 변수 x의 선언은 은폐(hidden)된다.
6. Static scoping rule은 변수 이름이 사용된 subprogram 내에서 찾고, 없으면, 그 subprogram을 호출한 subprogram에서 찾는다. 선언문을 찾을 때까지 이를 계속 반복하여, 호출된 subprogram을 역순으로 따라가며 찾는다.
7. Static scoping rule에서 subprogram의 중첩이 존재한다고 가정한다면 main() 속에 중첩되어 정의된 f()는 main()의 지역변수를 모두 접근할 수 있다.
8. subprogram에서 static으로 선언된 지역변수 (static 변수)의 scope과 존속기간은 변수 선언문부터 subprogram의 끝까지이다.
9. C++의 x가 전역변수라면 함수 내부에서는 $GLOBAL[‘x’]와 같이 접근된다.
10. Dynamic scoping rule는 호출 chain이 길면 실행 비용이 증가한다.
11. Static scoping rule이 적용되는 언어의 참조 환경 (referencing environment)은 지역변수, 정적 조상(static ancestors)의 지역변수(전역변수 포함)이며, 은폐된(hidden) 변수는 제외된다.

2. dynamic scoping rule의 단점을 두가지만 적고 이유를 간단히 적으시오.

|  |
| --- |
| program main;  var x : integer;  procedure sub1;  begin  writeln (‘x= ‘, x)  end ;  procedure sub2;  var x: integer;  begin  x:=5;  sub1  end;  begin  x:=10;  sub2  end. |

3. 오른쪽 프로그램이 static scoping rule일 때와 dynamic scoping rule 일 때 각각에 대해 sub1에서 출력되는 x의 값을 적으시오.

**Test #4. 2012.4.2 프로그래밍언어론 00 분반 조별 문제**

조번호:

참여자 (학번, 이름)

-

-

-

-

-

-

1. 다음 프로그램에서 counter()는 increment()라는 함수 정의를 리턴하는 함수이다.

def counter():

x = 0

def increment(y):

nonlocal x

x += y

print(x)

return increment

리턴되는 함수가 static scoping rule을 적용받는다고 할 때, 다음 프로그램이 출력하는 결과를 (유추해서?) 적어보시오.

counter1\_increment = counter()

counter2\_increment = counter()

counter1\_increment(1)

counter1\_increment(7)

counter2\_increment(1)

counter1\_increment(1)