**Test #8. 2013.4.30 프로그래밍언어론 02 분반**

학번:

이름:

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
2. call by name 은 lazy evaluation을 한다.
3. call by reference의 장점은 인자 전달 과정이 효율적이라는 것이다.
4. C에서 n개의 int 필드를 가지는 struct A를 리턴 타입으로 하는 함수를 호출하면, 리턴 값 전달 시간과 n은 무관하게 일정하다.
5. Python 에서 def foo(p1, p2, \*a, \*\*h); . 로 정의된 함수에 대해 foo(2, 4, 6, 8, mon=68, tue=72, wed=77) 로 호출하면, a에 해당하는 것은 [6,8]이 된다.
6. C#에서는 out 이라는 키워드를 통해 결과값을 callee 로 부터 전달받는 인자를 정의할 수 있다.

|  |
| --- |
| **procedure SUB1;**  var x: integer;  **procedure SUB2;**  begin  write('x =', x);  end;  **procedure SUB3;**  var x: integer;  begin  x := 3;  call sub4(SUB2);  end;  **procedure SUB4(subx);**  var x: integer;  begin  x := 4;  call subx;  end;  begin  x := 1;  call SUB3;  end; |

1. 오른쪽 프로그램에서 호출 순서가 sub1, sub3, sub4, sub2이고, 출력은 x=1였다면, 이 프로그램에서 call sub2가 실행될 때의 실행 환경은 전달된 subprogram을 (호출한/선언한/전달한) subprogram의 실행환경이다.
2. int foo(int f[][]) {…}과 같은 인자를 가지는 함수 선언이 C, C++에서 불가능한 이유를 간단히 적으시오.

답:

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
2. call by name 은 lazy evaluation을 한다. O
3. call by reference의 장점은 인자 전달 과정이 효율적이라는 것이다. O
4. C에서 n개의 int 필드를 가지는 struct A를 리턴 타입으로 하는 함수를 호출하면, 리턴 값 전달 시간과 n은 무관하게 일정하다. X
5. Python 에서 def foo(p1, p2, \*a, \*\*h); . 로 정의된 함수에 대해 foo(2, 4, 6, 8, mon=68, tue=72, wed=77) 로 호출하면, a에 해당하는 것은 [6,8]이 된다. O
6. C#에서는 out 이라는 키워드를 통해 결과값을 callee 로 부터 전달받는 인자를 정의할 수 있다. O

|  |
| --- |
| **procedure SUB1;**  var x: integer;  **procedure SUB2;**  begin  write('x =', x);  end;  **procedure SUB3;**  var x: integer;  begin  x := 3;  call sub4(SUB2);  end;  **procedure SUB4(subx);**  var x: integer;  begin  x := 4;  call subx;  end;  begin  x := 1;  call SUB3;  end; |

1. 오른쪽 프로그램에서 호출 순서가 sub1, sub3, sub4, sub2이고, 출력은 x=4였다면, 이 프로그램에서 call sub2가 실행될 때의 실행 환경은 전달된 subprogram을 (호출한/**선언한**/전달한) subprogram의 실행환경이다.
2. int foo(int f[][]) {…}과 같은 인자를 가지는 함수 선언이 C, C++에서 불가능한 이유를 간단히 적으시오.

C/C++ 에서 formal parameter가 배열일 때는

parameter로 전달 받은 배열의 크기 정보를 이용하여 local 변수처럼 선언되어야하는데

크기정보가 전달이 안 되므로 선언 자체가 불가능하다.