**Test #3. 2013.3.26 프로그래밍언어론 02 분반**

학번:

이름:

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
   1. x=10; 에서 10은 이 변수의 r-value이고 변수의 값을 나타낸다.
   2. Pointer를 사용하면 alias가 발생하나 reference 변수를 사용하면 발생하지 않는다.
   3. Implicit heap-dynamic 변수는 쓸모 없어 지면 보통 쓰레기 수집 (garbage collection)을 통해 반환된다.
   4. long f(long n) {int x=0; if (n<=0) return 1; else return f(x++)\*n; } 과

long f(long n) {**static** int x=0; if (n<=0) return 1; else return f(x++)\*n; } 의 결과는 같다.

* 1. 아래 코드에서 p는 stack dynamic variable 이다.

int foo()

{

int \*p;

p = (int\*)malloc(sizeof(int));

...

}

1. 타입을, 개발자가 정의하지 않고 컴파일러가 주변 상황으로부터 추론하는 것을 ( )라고 한다.

3. Dynamic type binding의 장,단점을 각 1개씩 적으시오.

답)

1. 다음 각 항목이 참이면 O, 아니면 X 를 하시오.
2. x=10; 에서 10은 이 변수의 r-value이고 변수의 값을 나타낸다. O
3. Pointer를 사용하면 alias가 발생하나 reference 변수를 사용하면 발생하지 않는다. X
4. Implicit heap-dynamic 변수는 쓸모 없어 지면 보통 쓰레기 수집 (garbage collection)을 통해 반환된다. O
5. long f(long n) {int x; if (n<=0) return 1; else return f(x)\*n; } 과

long f(long n) {**static** int x; if (n<=0) return 1; else return f(x)\*n; } 의 결과는 같다. X

1. 아래 코드에서 p는 stack dynamic variable 이다. O

int foo()

{

int \*p;

p = (int\*)malloc(sizeof(int));

...

}

1. 타입을, 개발자가 정의하지 않고 컴파일러가 주변 상황으로부터 추론하는 것을 ( type inference 타입 추론 )라고 한다.

3. Dynamic type binding의 장,단점을 각 1개이상 적으시오.

장 : 유연성(flexibility) 향상(, writability 향상)

단: type 오류를 발견하기 어려움, 프로그램 실행 비용 증가