

객체지향프로그래밍 - 10주차 실습 활동지 (2020년 11월 4일)

성명: _____ 학과: _____ 학번: _____ 실습실명: _____ 호

※ 본 실습활동지를 작성함에 있어 다른 학생의 문서로부터 일부 또는 전체를 복사하였습니까?

예() 아니오() (복사 하였다면 예에 체크하고 아니라면 아니오에 체크하시오)

제출방식: 별도의 답안지 파일에 각 문제에 대하여 작성한 답안이나 작성한 코드 및 그 실행 결과를 복사해 넣는다. 답안지 문서명은 "report_주차_학번" (예, report_10_201811111)로 만든다. 답안지는 pdf형식으로 과제게시판(Bb)에 차주 월요일 23시59분까지 제출한다.

I. Objectives

1. Lambda expression을 용도를 이해하고 작성할 수 있다.
2. Inner class의 용도를 이해하고 작성할 수 있다.

II. Review (15점)

1. String의 길이를 기준으로 다음 두가지 방식으로 정렬하는 코드를 작성하시오. 아래 코드를 완성할 것. (코드 및 수행결과 포함) (4점)

- 1) LengthComparator class를 정의하여 정렬
- 2) lambda expression을 이용하여 (1)번의 역순으로 정렬

```
class LengthComparator implements Comparator<String>
{ ... }
public class WordSortTest
{
    public static void main(String[] args)
    {
        String[] names = {"Lee", "Chang", "Hong"};
        // (1) sorting with LengthComparator
        // printing the sorted array
        // (2) sorting with lambda expression
        // printing the array sorted in reverse order
    }
}
```

2. 다음 코드는 구분구적법을 이용하여 적분을 계산하는 프로그램이다. 아래 프로그램을 수행해 본 후 물음에 답하시오.

```

interface F {
    double apply(double x);
}
public class IntegralTest
{
    public static double integral(F f, double a, double b, int n)
    {
        int i;
        double sum = 0;
        double dt = (b - a) / n;
        for(i = 0; i < n; i++)
        {
            sum += f.apply(a + (i + 0.5) * dt);
        }
        return sum*dt;
    }
    public static void main(String[] args)
    {
        F f;
        double r;
        f = x -> x*x;
        r = integral(f, 0, 10, 100);
        System.out.printf("f(x)=x*x, 0, 10, 100: %.5f%n", r);
    }
}

```

가) main 함수에서 다음 각 함수를 lambda expression으로 정의하여 적분 값을 구해보시오. 단, 괄호 속은 각각 a, b, n 값임. 소수점 이하 5번째 자리까지만 출력함. 힌트) printf를 이용. (2점)

- (1) $f(x) = x$ (0, 10, 100)
 (2) $f(x) = \sin(x)$ (0, π , 100) 단, π 는 원주율

나) 0~ π 구간의 $\sin(x)$ 적분 값은 2이지만 (가)에서 구한 값은 2의 근사치가 나온다. 2가 되는 n의 최소 값을 구해보시오. 단, 소수점 이하 5번째 자리까지만 사용. (2점) (코드와 실행결과 포함)

3. 강의노트 ch06-part5의 Listing 6.7을 수행해본 후 다음 물음에 답하시오.

가) TalkingClock의 start 메소드 내에서 TimePrinter 객체를 생성하여 Timer 객체에 등록하는 작업을 InnerClassTest class의 main 함수에서 하도록 변경하여 동일한 결과가 나오도록 수정하시오. (코드 및 수행화면) [3점]

힌트) 강의노트 ch06-part5의 Special Syntax Rule을 참조할 것

4. 강의노트 ch06-part5의 Listing 6.8의 anonymous inner class를 수행해본 후 다음 두 가지 방식으로 수정하여 실행해보시오. (코드 및 수행화면) [4점]

가) local inner class (2점)

나) lambda expression (2점)