

Instructions. You will have 50 minutes to create and submit one program and a readme file. Debug your code as needed. You may use your book, your notes, your code from programming assignments, the code on the COS 126 course website, the booksite, and you may read Piazza. No form of communication is permitted (e.g., talking, email, texting, calling, posting to Piazza) during the exam.

Submissions. Submit your work using the link on the Meetings page.

Grading. Your program will be graded on correctness, design, efficiency, and clarity including comments. You will receive partial credit for a program that correctly implements some of the required functionality. You will lose a large number of points if your program does not compile.

Discussing this exam. Due to travel for extracurriculars and sports, some of your peers will take this exam next week. Do not discuss exam contents with anyone who has not taken the exam.

This paper. In addition to submitting your code electronically, you must return this paper. Fill in the information below, then transcribe and sign the Honor Code pledge. You may do so now.

NAME: _____

NETID: _____

PRECEPT: _____

EXAM ROOM: _____

"I pledge my honor that I will not violate the Honor Code during this examination."

SIGNATURE: _____

Write a program, `DrawPath.java`, that draws line segments in the standard drawing window based on data read from standard input. Start with the provided template, found on the Meetings page:

```
public class DrawPath {
    public static void main(String[] args) {

        // starting point
        double x = StdIn.readDouble();
        double y = StdIn.readDouble();

        // YOUR CODE HERE

    }
}
```

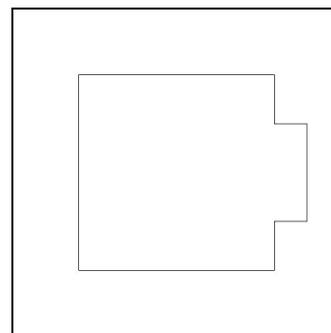
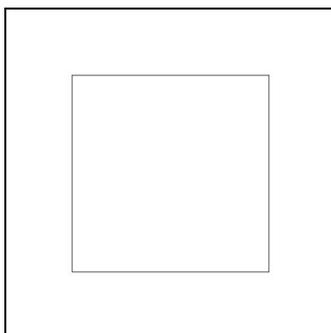
The input to this program consists of pairs of values, where the first pair is the starting point (x, y) , and all subsequent pairs are displacements, (dx, dy) . You may assume that all input files will have at least four values and have an even number of values (i.e., for every dx , there will be a dy). For example, using the inputs in `pathA.txt` as listed in the figure on the right, the starting point is $(.2, .2)$ and the first (dx, dy) is $(.6, 0)$, so the first line segment will connect $(.2, .2)$ to $(.8, .2)$, then $(.8, .2)$ to $(.8, .8)$ and so on.

`pathA.txt`

.2	.2
.6	0
0	.6
-.6	0
0	-.6

Under "YOUR CODE HERE", write a single loop that reads double values dx and dy from `StdIn`, uses `StdDraw.line()` to draw a line segment from (x, y) to $(x + dx, y + dy)$ and then updates x to $x + dx$ and y to $y + dy$. Do not use `StdDraw.polygon()`. Note, use of unnecessary data structures will be penalized.

Your code should produce the following drawings for the input files `pathA.txt` and `pathB.txt`:



> `java-introcs DrawPath < pathA.txt`

> `java-introcs DrawPath < pathB.txt`

This part of the exam is worth 2/3rds of your grade. Complete this part before moving on to the next part.

