# Digital Pakistan Speed Programing Competition Online Qualifier Round

**Instructions**

- Do not open the booklet unless you are explicitly told to do so. You can only read these instructions below.
- If you have any question regarding the problems, seek a clarification from the judges using DOMJudge.
- Before submitting a run, make sure that it is executable via command line. For Java, it must be executable via "javac" and for GNU C++ via "g++". Java programmers need to remove any "package" statements and source code's file name must be the same as of main class. C++ programmers need to remove any getch() / system("pause") like statements.
- Do not attach input files while submitting a run, only submit/attach source code files, i.e., *.java or *.cpp or *.py.
- Language supported: C/C++, Java and Python3
- Source code file name should not contain white space or special characters.
- You must take input from Console i.e.: Standard Input Stream (stdin in C, cin in C++, System.in in Java, stdin in Python)
- You must print your output to Console i.e.: Standard Output Stream (stdout in C, cout in C++, System.out in Java)
- Please, don't create/open any file for input or output.
- Please strictly meet the output format requirements as described in problem statements, because your program will be auto judged by computer. Your output will be compared with judge's output byte-by-byte and not tolerate even a difference of single byte. So, be aware! **Pay special attention to spaces, commas, dots, newlines, decimal places, case sensitivity etc.**
- All your programs must meet the time constraint specified.
- The decision of judges will be absolutely final.

## Problem 04: Perfect Cover Drive

**Time limit: 1 second**

In the sport of cricket, a batter aims to place the ball in the gaps between fielders to score runs. You are given the 2D top-down view of a cricket field, where the batter stands at the origin (0, 0). The batter can hit the ball in **any direction**, and the ball will travel in a straight line indefinitely (no bounce or friction). There are **n** fielders on the field, each at a fixed position. Each fielder has a **circular range of influence** with radius **r**, within which they can stop the ball.

Your task is to determine the **maximum angle of a gap** between any two fielders (i.e., an angular sector not covered by any fielder's circle), such that the batter can safely hit the ball through that sector without it being intercepted.

**Input Format**

- The first line contains two values - an Integer **n** $(1 \leq n \leq 11)$ denoting the number of fielders, and a Float **r** $(0 < r \leq 10^5)$ denoting the radius of influence of each fielder.
- **n** subsequent lines follow: each with two floats $\mathbf{x_i}$ $\mathbf{y_i}$ $(-100 \leq \mathbf{x_i}, \mathbf{y_i} \leq 100)$ denoting the coordinates of the $\mathrm{i}^{\mathrm{th}}$ $(1 \leq i \leq n)$fielder.

**Output Format**

- A float: The **maximum angular gap** (in degrees) not covered by any fielder, with **at least 6 digits after the decimal**.

**Constraints**

- Fielders may be placed in such a way that their regions overlap.
- You may assume fielders are not located at (0, 0).
- If the field is completely covered, output 0.000000.

| **Sample input** | **Sample Output** |
|---|---|
| 2 1.0<br>1.0 0.0<br>-1.0 0.0 | 0.000000 |
| 3 4.5<br>-4.5 7.794<br>-9.0 0.0<br>-4.5 -7.794 | 180.000000 |