

[Dashboard](#) / [My courses](#) / [ITB IF2110 1 2425](#) / [Praktikum 1](#) / [ADT Sederhana - Latihan Praktikum](#)

Started on	Sunday, 29 September 2024, 10:12 PM
State	Finished
Completed on	Sunday, 29 September 2024, 10:48 PM
Time taken	35 mins 23 secs
Grade	300.00 out of 300.00 (100%)

Question **1**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Download file header ADT Lingkaran dalam [circle.h](#) dan buatlah file implementasinya. Kumpulkan hanya file **circle.c**.

Catatan:

Gunakan Implementasi ADT Point dari Pra-Praktikum!

Buatlah driver sendiri untuk mengetes setiap fungsi/prosedur yang ada!

C

 [circle.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	5	Accepted	0.00 sec, 1.71 MB
2	5	Accepted	0.00 sec, 1.66 MB
3	5	Accepted	0.00 sec, 1.66 MB
4	5	Accepted	0.00 sec, 1.71 MB
5	5	Accepted	0.00 sec, 1.50 MB
6	5	Accepted	0.00 sec, 1.54 MB
7	5	Accepted	0.00 sec, 1.73 MB
8	5	Accepted	0.00 sec, 1.65 MB
9	5	Accepted	0.00 sec, 1.79 MB
10	5	Accepted	0.00 sec, 1.63 MB
11	5	Accepted	0.00 sec, 1.85 MB
12	5	Accepted	0.00 sec, 1.80 MB
13	5	Accepted	0.00 sec, 1.79 MB
14	5	Accepted	0.00 sec, 1.68 MB
15	5	Accepted	0.00 sec, 1.84 MB
16	5	Accepted	0.00 sec, 1.75 MB
17	5	Accepted	0.00 sec, 1.80 MB
18	5	Accepted	0.00 sec, 1.83 MB

No	Score	Verdict	Description
19	5	Accepted	0.00 sec, 1.75 MB
20	5	Accepted	0.00 sec, 1.80 MB

Question **2**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Transformasi Titik Dengan Bilangan Kompleks

Anda diberikan sebuah titik $P(x,y)$ di dalam bidang kartesian, serta bilangan kompleks $C = a+bi$. Tugas Anda adalah mengaplikasikan transformasi iteratif terhadap titik P , di mana pada setiap iterasi, titik P akan diputar dan diperbesar menggunakan bilangan kompleks C .

Transformasi ini dilakukan sebanyak n kali, di mana setiap iterasi melibatkan penghitungan hasil perpangkatan bilangan kompleks C_i , kemudian menggunakan hasil perpangkatan tersebut untuk memutar titik P dan memperbesarnya. Pada setiap iterasi, Anda juga perlu memeriksa apakah titik P tetap berada di dalam lingkaran satuan ($x^2 + y^2 \leq 1$). Jika titik keluar dari lingkaran satuan, iterasi dihentikan dan program mencetak pada iterasi ke berapa titik keluar dari lingkaran.

Jadi, prosedurnya adalah sebagai berikut:

1. Hitung nilai $pC = C^i$, ($0 \leq i \leq n$), i adalah jumlah iterasi saat ini
- 2a. Ambil nilai x baru dengan rumus: nilai point $X * \text{Re}(pC) - \text{nilai point } Y * \text{Im}(pC)$
- 2b. Ambil nilai y baru dengan rumus: nilai point $X * \text{Im}(pC) + \text{nilai point } Y * \text{Re}(pC)$
3. Tentukan apakah titik (x, y) baru melebihi satuan lingkaran ($x^2 + y^2 > 1$, x dan y adalah nilai (x,y) titik baru setelah transformasi).

Implementasikan fungsi **PowerCOMPLEX** yang menerima bilangan kompleks C jumlah pangkat p serta **TransformPointByComplexPower** yang menerima titik P , bilangan kompleks C , dan jumlah iterasi n . Berikut adalah file headernya [rotating_point.h](#). Implementasikan fungsi-fungsi ini dalam file **rotating_point.c**.

Catatan

Gunakan Implementasi ADT Bilangan Kompleks dan ADT Point dari Pra-Praktikum!

- $\text{Re}(C)$ adalah elemen bilangan real dari bilangan kompleks C
- $\text{Im}(C)$ adalah elemen bilangan imajiner dari bilangan kompleks C
- Contoh: $C = 3+2i$, $\text{Re}(C) = 3$, $\text{Im}(C) = 2$

Contoh

No	Masukan	Luaran
1.	0.5 0.5 1.0 0.5 2	Titik tetap berada di dalam lingkaran setelah 2 iterasi
2.	0.5 0.5 1.5 0.5 2	Titik keluar dari lingkaran pada iterasi ke 1
x.	Px Py Re(C) Im(C) n	Px nilai x pada point Py nilai y pada point n adalah jumlah iterasi

Penjelasan **No.1**

- Titik awal adalah $P(0.5, 0.5)$, nilai $x^2 + y^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5$, artinya titik masih berada di dalam lingkaran satuan ($x^2 + y^2 \leq 1$)
- Iterasi 1:
 - Nilai $C^1 = 1.00 + 0.50i$
 - Maka nilai titik X baru adalah $0.5 * 1 - 0.5 * 0.5 = 0.25$
 - Maka nilai titik Y baru adalah $0.5 * 0.5 + 0.5 * 1 = 0.75$
 - Maka nilai titik baru adalah $P'(0.25, 0.75)$
- Iterasi 2:
 - Nilai $C^2 = 0.75 + 1.00i$
 - Menggunakan cara yang sama seperti pada iterasi 1, nilai titik baru adalah $P''(-0.56, 0.81)$, yang nilai $x^2 + y^2 = (-0.56)^2 + (0.81)^2 = 0.9697$
 - Karena iterasi selesai dan nilai satuan lingkarannya masih ≤ 1 , maka luaran yang dikeluarkan adalah "Titik tetap berada di dalam lingkaran setelah 2 iterasi", tanpa tanda petik

C

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	10	Accepted	0.00 sec, 1.67 MB
2	10	Accepted	0.00 sec, 1.73 MB
3	10	Accepted	0.00 sec, 1.55 MB
4	10	Accepted	0.00 sec, 1.68 MB
5	10	Accepted	0.00 sec, 1.70 MB
6	10	Accepted	0.00 sec, 1.68 MB
7	10	Accepted	0.00 sec, 1.71 MB
8	10	Accepted	0.00 sec, 1.63 MB
9	10	Accepted	0.29 sec, 1.68 MB
10	10	Accepted	0.00 sec, 1.59 MB

Question **3**

Correct

Mark 100.00 out of 100.00

Time limit	1 s
Memory limit	64 MB

Download file header ADT Lingkaran dalam [fraction.h](#) dan buatlah file implementasinya. Kumpulkan hanya file **fraction.c**.

Catatan:

Buatlah driver sendiri untuk mengetes setiap fungsi/prosedur yang ada!

C

 [fraction.c](#)

Score: 100

Blackbox

Score: 100

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	8	Accepted	0.00 sec, 1.73 MB
2	8	Accepted	0.00 sec, 1.68 MB
3	8	Accepted	0.00 sec, 1.67 MB
4	8	Accepted	0.00 sec, 1.55 MB
5	8	Accepted	0.00 sec, 1.63 MB
6	8	Accepted	0.00 sec, 1.63 MB
7	8	Accepted	0.00 sec, 1.60 MB
8	8	Accepted	0.00 sec, 1.63 MB
9	8	Accepted	0.00 sec, 1.64 MB
10	8	Accepted	0.00 sec, 1.70 MB
11	10	Accepted	0.00 sec, 1.50 MB
12	10	Accepted	0.00 sec, 1.60 MB

◀ [ADT Sederhana - Praktikum](#)

Jump to...

[List Statik - Pra Praktikum](#) ►