

Nama : Luthfi Arie Zulfikri

NIM : H1D022061

Shift KRS : A

Shift Baru : A

## **TUGAS 1 – PRAKTIKUM PEMROGRAMAN MOBILE PERMAINAN TIC TAC TOE**

### **Penjelasan Kode**

#### **1. Inisialisasi**

Melakukan inisialisasi data awal papan permainan, kemudian fungsi `playGame()` dipanggil untuk memulai permainan.

#### **2. Memulai Permainan**

Variabel `play` diatur ke `true` untuk menandakan bahwa permainan dimulai serta variabel `count` diinisialisasi sebagai penghitung giliran pemain. Loop utama permainan dimulai, di mana pemain akan bergantian bermain hingga ada pemenang atau permainan seri.

#### **3. Memasukkan Input Pemain (`takeInp()`)**

Fungsi `takeInp()` dipanggil untuk menerima input dari pemain. Fungsi ini pertama-tama mencetak papan permainan saat ini. Program kemudian meminta pemain memasukkan angka (1-9) untuk menentukan posisi di papan yang ingin diisi.

#### **4. Memvalidasi Input (`readInp()` dan `checkData()`)**

Input pemain (angka 1-9) diproses oleh fungsi `readInp()` dengan menerjemahkan input pemain menjadi indeks baris dan kolom yang sesuai di papan 3x3. Setelah posisi dipetakan, fungsi `checkData()` digunakan untuk memeriksa apakah kotak pada posisi tersebut sudah diisi dengan "X" atau "O". Jika kotak sudah diisi, pemain diminta untuk mengulang input. Jika kotak kosong, simbol pemain (dikonversi dari angka 0 atau 1 menjadi "X" atau "O") ditempatkan di papan pada posisi yang dipilih.

#### **5. Pemeriksaan Kondisi Menang (`checkWinner()`)**

Setelah pemain mengisi posisi di papan, program memeriksa apakah pemain tersebut menang dengan fungsi `checkWinner()` yang memeriksa semua baris, kolom, dan dua diagonal untuk melihat apakah ada tiga simbol yang sama (baik "X" atau "O") secara

berturut-turut. Jika pemain menang, papan permainan dicetak lagi dan program menampilkan pesan kemenangan dan permainan berakhir dengan menghentikan loop.

#### 6. Pemeriksaan Kondisi Seri (checkDraw())

Program memeriksa apakah papan sudah penuh (semua kotak terisi "X" atau "O") menggunakan fungsi checkDraw(). Jika papan penuh dan tidak ada pemenang, permainan dianggap seri dan permainan berakhir.

#### 7. Mengulang Giliran

Jika tidak ada pemenang dan permainan belum seri, variabel count bertambah, sehingga giliran berpindah ke pemain berikutnya. Loop berlanjut dengan memanggil takeInp() lagi untuk menerima input dari pemain berikutnya.

#### 8. Mengakhiri Permainan

Setelah ada pemenang atau hasil seri, loop utama di dalam playGame() dihentikan. Program selesai dan tidak menerima input lagi dari pemain.