# LAPORAN UJIAN TENGAH SEMESTER KOMPUTER DAN PEMROGRAMAN



## DISUSUN OLEH :

Kelompok 8

1. Adelia Nurazizah Omega Putri (G1A023022)
2. Khalisa Rizgita Amanda (G1A023080) Kelas : B Informatika

NAMA ASISTEN DOSEN

Randi Julian Saputra (G1A019066)

Dosen Pengampu

Arie Vatresia, S.T. M.TI., P.hD

# PROGRAM STUDI INFORMATIKA FAKULTAS TEKNIK UNIVERSITAS BENGKULU

**2023**

# LANDASAN TEORI

Java adalah nama sebuah bahasa pemrograman yang sangat terkenal.Sebagai Bahasa pemrograman, Java dapat digunakan untuk menulis program.Sebagaimana diketahui, program adalah Kumpulan instruksi yang ditujukan untuk komputer.Melalui program, computer dapat diatur agar melaksanakan tugas tertentu sesuai yang ditentukan oleh pemrogram (orang yang membuat program).Sebagai contoh, Anda mungkin terbiasa menggunakan pengolah kata seperti Word.Nah, Word adalah contoh aplikasi yang dibuat dengan menggunakan Bahasa pemrograman.

Bahasa Java dikembangkan di Sun Microsystems yang dimulai oleh James Gosling, dan mulai diperkenalkan kepada publik pada tahun 1995.Seperti halnya C++, Java juga merupakan Bahasa yang berorientasi objek.Dengan demikian, Java juga memudahkan dalam pembuatan aplikasi yang berskala besar.

Saat ini Sun Microsystem telah diakuisisi ole Oracle Corporatian.Java bersifat Write Once, Run Anywhere (program yang ditulis satu kali dan dapat berjalan pada banyak platform).

Berikut in fitur-fitur Java:

* Berorientasi Objek: Dalam Java, semua adalah Objek.
* Bersifat Platform Independent: Java di-compile dalam bit kode platform independen dan bukan pada mesin platform spesifik seperti pada C dan C+ +.
* Sederhana: Java didesain untuk dapat dengan mudah dipelajari.
* Aman: Dengan fitur keamanan Java, Anda dapat membuat sistem yang bebas virus dan powerful.
* Bersifat Architectural-neutral: Compiler Java membuat format file objek yang architectural-neutral, yang membuat kode yang decompile dapat dieksekusi pada berbagai prosesor yang memiliki sistem runtime Java.
* Portabel: Java bersifat portable karena adanya fitur platform independent dan architectural-neutral.
* Kuat dan powerful: Java mengeliminasi error dengan menjalankan pengecekan pada waktu compile dan runtime.
* Multithreaded: Dengan fitur multithread Java, Anda dapat membuat program yang dapat mengerjakan banyak tugas sekaligus.
* Terinterpretasi: Kode bit Java ditranslasi secara langsung pada instruksi mesin dan tidak disimpan.
* Performa tinggi: Java memiliki performa yang tinggi karena menggunakan compiler langsung.
* Terdistribusi: Java didesain untuk lingkungan distribusi internet.
* Dinamis: Java lebih dinamis dari C dan C+ + karena Java didesain untuk beradaptasi dengan lingkungan pengembangan.

Bahasa pemrograman java berbasis objek, yang berarti hampir semua elemen dalam bahasa ini adalah objek. Ini memungkinkan pengembang untuk membuat kode yang mudah dimengerti, modular, dan dapat digunakan kembali. Salah satu fitur utama Java adalah kemampuannya untuk berjalan di berbagai platform (Platform Independence). Hal ini dicapai dengan menggunakan JVM (Java Virtual Machine), yang menerjemahkan kode Java menjadi bytecode yang dapat dijalankan di berbagai sistem operasi tanpa perlu melakukan perubahan pada kode sumber.

Software ini memiliki berbagai mekanisme keamanan seperti sandboxing dan pengendalian akses yang ketat yang membantu melindungi sistem dari ancaman keamanan. Kode Java dapat dengan mudah dipindahkan dari satu platform ke platform lainnya tanpa perlu melakukan banyak perubahan, Ini menjadikannya bahasa yang ideal untuk pengembangan perangkat lunak lintas platform.

Java menggunakan pengumpulan sampah (garbage collection) untuk mengelola memori. berarti pengembang tidak perlu secara eksplisit mengelola alokasi dan pembebasan memori. Java dilengkapi dengan libreri standar yang kaya yang mencakup berbagai fungsi dan kelas yang dapat digunakan pengembang untuk mempercepat pengembangan aplikasi. Juga memiliki dukungan bawaan untuk pemrograman berbasis multithreading, yang memungkinkan aplikasi untuk menjalankan beberapa tugas secara bersamaan. Ini berguna untuk meningkatkan kinerja dan responsifitas aplikasi.

Setelah dirilis pada tahun 1995, Java segera mendapatkan popularitas karena keunggulan dalam portabilitasnya. Kode Java dapat dijalankan di berbagai platform tanpa perlu dimodifikasi ulang, berkat penggunaan Java Virtual Machine (JVM). Selama periode ini, Java mengalami pertumbuhan pesat dalam penggunaan dan pengembangannya, terutama di aplikasi web dan server.

Pada tahun 2010, Oracle Corporation mengakuisisi Sun Microsystems, dan dengan itu, Java menjadi salah satu produk unggulan Oracle. Oracle melanjutkan pengembangan Java, merilis versi-versi baru dengan peningkatan kinerja, keamanan, dan fitur-fitur baru. Java dibagi menjadi beberapa edisi untuk memenuhi kebutuhan berbagai jenis aplikasi.

1. Java Standard Edition (SE) digunakan untuk pengembangan aplikasi desktop dan server.
2. Java Enterprise Edition (EE) digunakan untuk pengembangan aplikasi web dan enterprise.
3. Java Micro Edition (ME) digunakan untuk pengembangan aplikasi di perangkat mobile dan embedded.

Pada tahun 2010, Oracle Corporation mengakuisisi Sun Microsystems, sehingga mengambil alih pengembangan dan dukungan untuk Java. Java tetap menjadi bahasa pemrograman yang sangat populer di seluruh dunia. Java juga memainkan peran kunci dalam pengembangan sistem operasi mobile Android. Android menggunakan bahasa pemrograman Java dalam pengembangan aplikasi mobile, yang membuatnya menjadi salah satu platform mobile yang paling populer di dunia.

Java terus berkembang dengan versi-versi terbaru yang menambahkan fitur-fitur baru dan penyempurnaan.

Saat ini Java telah menjadi bahasa pemrograman yang sangat berpengaruh dalam industri perangkat lunak dan digunakan dalam berbagai jenis proyek, termasuk aplikasi desktop, aplikasi web, perangkat lunak server, perangkat mobile, dan banyak lagi. Keunggulan seperti portabilitas, keamanan, dan dukungan yang luas telah menjadikan Java salah satu bahasa pemrograman yang paling banyak digunakan di dunia.dan di sini belajar membuat game tic tac toe serta menggunakan [rpgra, java yang sangat menarik untuk di pelajari dan implementasiikan dalam pembelajaran bahasa coding khususnya ,

## SOAL DAN PEMBAHASAN

1. Buatlah Game Tictactoe atau Turtle Maze menggunakan java :
   * Library

Printscreen

Kelompok 8



Gambar 1.1 Source code library

Penjelasan Source code:

Kode di atas digunakan untuk menggunakan pustaka (library) AWT (Abstract Window Toolkit) dan Swing untuk membuat antarmuka grafis (GUI) sederhana. Kode ini membuat sebuah jendela (window) dengan sebuah tombol (button) di dalamnya.

* + Variable

Printscreen

Kelompok 8



Gambar 1.2 Source code Variable

Penjelasan Source code:

Kode di atas adalah deklarasi variabel-variabel dan objek-objek yang akan digunakan dalam permainan Tic Tac Toe kali ini.

* + Antarmuka Grafis

Printscreen

Kelompok 8



Gambar 1.3 Source code

Penjelasan source code:

Frame permainan diatur dengan ukuran 700x700 piksel dan ditutup ketika tombol close ditekan. Judul permainan ("Tic-Tac-Toe") ditampilkan dalam TextField dengan latar belakang hitam dan teks hijau terang, menggunakan font "Ink Free" dengan ukuran 75. Dua panel dibuat: ‘titlePanel’ menggunakan layout ‘BorderLayout’ dan ‘buttonPanel’ menggunakan ‘GridLayout 3x3’. Latar belakang buttonPanel diatur menjadi abu-abu gelap. Sembilan tombol dibuat dan ditambahkan ke buttonPanel. Setiap tombol menggunakan font "Mv Boli" dengan ukuran 120 dan tidak dapat difokuskan oleh pengguna. ‘TextField’ judul ditempatkan di bagian atas frame,

sementara tombol-tombol ditempatkan di tengah frame menggunakan layout ‘BorderLayout’. ‘startGame()’ menginisialisasi logika permainan Tic-Tac-Toe seperti inisialisasi papan permainan, mengatur giliran pemainPrintscreen

* + Tombol Pada Permainan

Printscreen

Kelompok 8



Gambar 1.4 Source code

Penjelasan Source code:

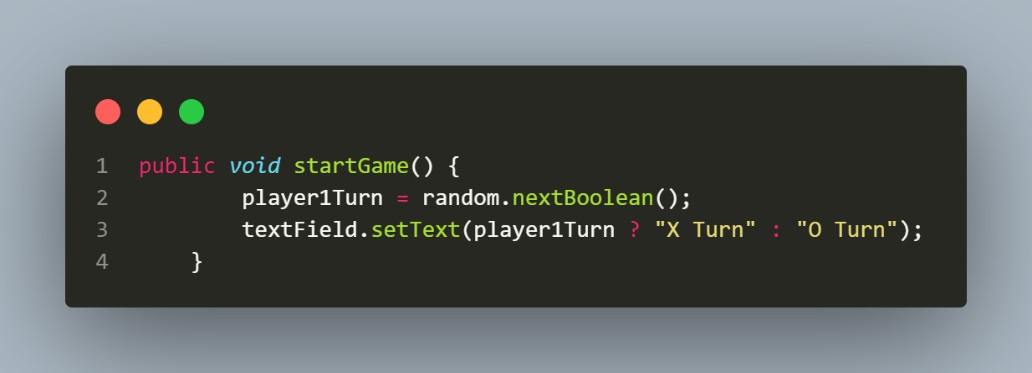
Kode di atas adalah implementasi dari metode ‘actionPerformed(ActionEvent e)’, yang merupakan bagian dari antarmuka ‘ActionListener’. Metode ini dipanggil ketika suatu aksi terjadi, seperti saat pengguna mengklik tombol pada permainan Tic-Tac-Toe.Memerikasa Papan Penuh atau Belum. Loop Iterasi kode ini menggunakan loop for untuk mengiterasi melalui array buttons yang mewakili tombol-tombol permainan (indeks 0 hingga 8). ‘e.getSource() == buttons[i]’ memeriksa apakah aksi berasal dari tombol ke-i. ‘buttons[i].getText().equals("")’ memeriksa apakah tombol tersebut belum diisi (memiliki teks kosong). ‘!gameOver’ memastikan bahwa permainan masih berlangsung. Jika giliran pemain ‘1 (player1Turn bernilai true)’, teks tombol diubah menjadi "X" dengan warna merah. Jika giliran pemain 2 ‘(player1Turn bernilai false)’, teks tombol diubah menjadi "O" dengan warna biru. Setelah mengubah teks tombol, giliran pemain dibalikkan dengan ‘player1Turn =

!player1Turn’. Label ‘textField’ diperbarui dengan pesan yang menunjukkan giliran pemain saat ini ‘("X Turn" atau "O Turn")’. Setelah setiap langkah, metode ‘check()’ dipanggil. Fungsi ini mungkin digunakan untuk memeriksa apakah ada pemenang atau apakah permainan berakhir seri. Implementasi ‘check()’ tidak disediakan dalam potongan kode yang diberikan.

* + Implementasi Dari startGame()

Printscreen

Kelompok 8



Gambar 1.5 Source code

Penjelasan Source code:

‘player1Turn = random.nextBoolean();’ kode ini menggunakan objek random untuk menghasilkan nilai boolean acak (‘true’ atau ‘false’). Jadi, ‘player1Turn’ akan memiliki nilai true atau false secara acak. Ini menentukan giliran pemain pertama. Jika ‘player1Turn’ bernilai ‘true’, pemain pertama (X) akan bermain duluan. Jika ‘player1Turn’ bernilai ‘false’, pemain kedua (O) yang akan bermain duluan. ‘textField.setText(player1Turn ? "X Turn" : "O Turn");’ kode ini mengatur teks pada ‘textField’ (label) berdasarkan nilai ‘player1Turn’. Jika ‘player1Turn’ adalah ‘true’ (pemain pertama bermain duluan), label akan menampilkan "X Turn". Jika ‘player1Turn’ adalah ‘false’ (pemain kedua bermain duluan), label akan menampilkan "O Turn".Memeriksa apakah permainan tic-tac-toe telah berakhir.

* + Implementasi Dari check()

Printscreen

Kelompok 8



Gambar 1.6 Source code

Penjelasan Source code:

‘for (int[] winningPosition : winningPositions) { ... }’ kode ini menggunakan perulangan foreach untuk mengiterasi melalui setiap posisi kemenangan yang telah ditentukan sebelumnya. Setiap ‘winningPosition’ adalah array tiga elemen yang mewakili indeks tombol pada ‘buttons’ yang membentuk sebuah baris, kolom, atau diagonal di papan permainan. ‘buttons[winningPosition[0]].getText().equals(buttons[winningPosition[1]].getText())’ kode ini membandingkan teks (X atau O) pada dua tombol pertama dari ‘winningPosition’ untuk melihat apakah mereka sama. ‘buttons[winningPosition[1]].getText().equals(buttons[winningPosition[2]].getText())’ kode ini membandingkan teks pada dua tombol kedua dari ‘winningPosition’ untuk memastikan bahwa semua tiga tombol di ‘winningPosition’ memiliki teks yang sama (entah semuanya X atau semuanya O). ‘!buttons[winningPosition[0]].getText().equals("")’ kode ini memeriksa apakah teks pada tombol pertama dari ‘winningPosition’ tidak kosong. Ini memastikan bahwa tiga tombol tersebut telah diisi oleh pemain (entah X atau O) dan tidak kosong. ‘gameOver = true;’ jika kondisi di atas terpenuhi, variabel ‘gameOver’ diatur menjadi ‘true’, menandakan bahwa permainan telah berakhir. ‘highlightWinner(winningPosition);’ metode ‘highlightWinner(winningPosition)’ kemungkinan besar berfungsi untuk menyorot (mengganti warna, memberi efek visual, dll.) tombol-tombol pada ‘winningPosition’ untuk menunjukkan baris, kolom, atau diagonal yang memenangkan permainan. ‘updateStatus(buttons[winningPosition[0]].getText() + " Wins!");’ metode ‘updateStatus(String message)’ kemungkinan berfungsi untuk memperbarui pesan status pada antarmuka pengguna. Dalam konteks ini, pesan yang diperbarui akan menampilkan pemain yang memenangkan permainan (X atau O) dengan tambahan kata "Wins!".

* + Menentukan Apakah Permainan Berakhir Seri

Printscreen

Kelompok 8



Gambar 1.7 Source code

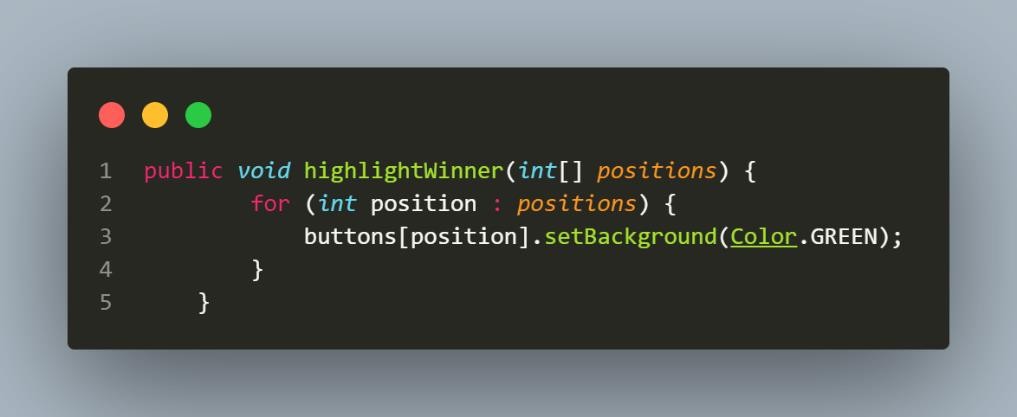
Penjelasan Source code:

‘if (!gameOver) { ... }’ kode ini memeriksa apakah permainan belum berakhir (‘gameOver’ adalah ‘false’). Jika permainan sudah berakhir (karena ada pemenang), langkah-langkah berikutnya tidak akan dieksekusi. ‘boolean allButtonsClicked = true;’ variabel ‘allButtonsClicked’ diinisialisasi sebagai ‘true’. Ini menandakan bahwa semua tombol permainan telah diklik oleh pemain. ‘for (JButton button : buttons) { ... }’ kode ini menggunakan perulangan ‘foreach’ untuk mengiterasi melalui array ‘buttons’ (semua tombol permainan). ‘if (button.getText().equals("")) { allButtonsClicked = false; break; }’ dalam setiap iterasi, kode memeriksa apakah teks tombol saat ini kosong (""). Jika ada setidaknya satu tombol yang masih kosong, ‘allButtonsClicked’ diatur menjadi ‘false’ dan perulangan dihentikan dengan menggunakan ‘break’. Ini berarti tidak semua tombol telah diklik. ‘if (allButtonsClicked) { ... }’ Setelah iterasi selesai, kode memeriksa apakah semua tombol telah diklik (artinya tidak ada yang kosong). Jika itu terjadi, variabel ‘gameOver’ diatur menjadi ‘true’ untuk menunjukkan bahwa permainan telah berakhir. Pesan status diperbarui dengan teks "It's a draw!" untuk memberitahu pengguna bahwa permainan berakhir seri.

* + Implementasi Metode highlightWinner(int[] positions)

Printscreen

Kelompok 8



Gambar 1.8 Source code

Penjelasan Source code:

‘public void highlightWinner(int[] positions) { ... }’ Ini adalah definisi dari metode ‘highlightWinner’. Metode ini mengambil satu parameter, yaitu array ‘positions’ yang berisi indeks-indeks tombol yang harus disorot karena membentuk baris, kolom, atau diagonal kemenangan. ‘for (int position : positions) { ... }’ kode ini menggunakan perulangan foreach untuk mengiterasi melalui setiap elemen dalam array ‘positions’.

‘buttons[position].setBackground(Color.GREEN);’ dalam setiap iterasi, kode mengakses tombol pada indeks yang diberikan oleh ‘position’ dan mengubah warna latar belakangnya menjadi hijau (‘Color.GREEN’). Dengan mengubah warna latar belakang tombol-tombol ini menjadi hijau, mereka disorot dan menunjukkan baris, kolom, atau diagonal yang memenangkan permainan.

* + Memperbarui Pesan Status Pada Antarmuka Permainan

Printscreen

Kelompok 8



Gambar 1.9 Source code

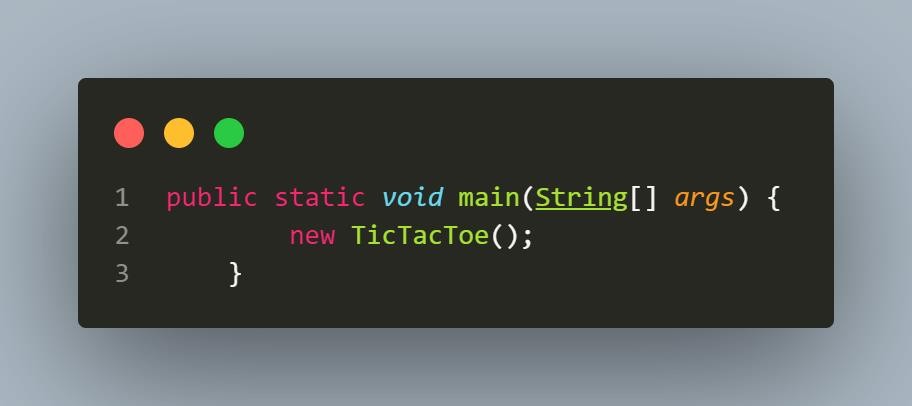
Penjelasan Source code:

‘public void updateStatus(String message) { ... }’ Ini adalah definisi dari metode ‘updateStatus’. Metode ini mengambil satu parameter, yaitu ‘message’, yang adalah pesan yang akan ditampilkan pada antarmuka permainan untuk memberi tahu pemain tentang hasil permainan (kemenangan, seri, atau pesan lainnya). ‘for (JButton button : buttons) { ... }’ kode ini menggunakan perulangan foreach untuk mengiterasi melalui setiap tombol permainan yang ada dalam array ‘buttons’. ‘button.setEnabled(false);’ dalam setiap iterasi, kode menonaktifkan tombol dengan mengatur ‘setEnabled(false)’. Ini berarti tombol-tombol permainan tidak dapat diklik lagi setelah permainan berakhir. Pemain tidak dapat melakukan langkah tambahan setelah permainan selesai. ‘textField.setText(message);’ setelah menonaktifkan tombol-tombol permainan, kode mengatur teks pada ‘textField’ (label status) dengan pesan yang diberikan dalam parameter ‘message’. Pesan ini akan memberitahu pemain tentang hasil permainan, seperti siapa yang memenangkan permainan atau bahwa permainan berakhir seri.

* + Memanggil Tic Tac Toe

Printscreen

Kelompok 8



Gambar 1.10 Source code

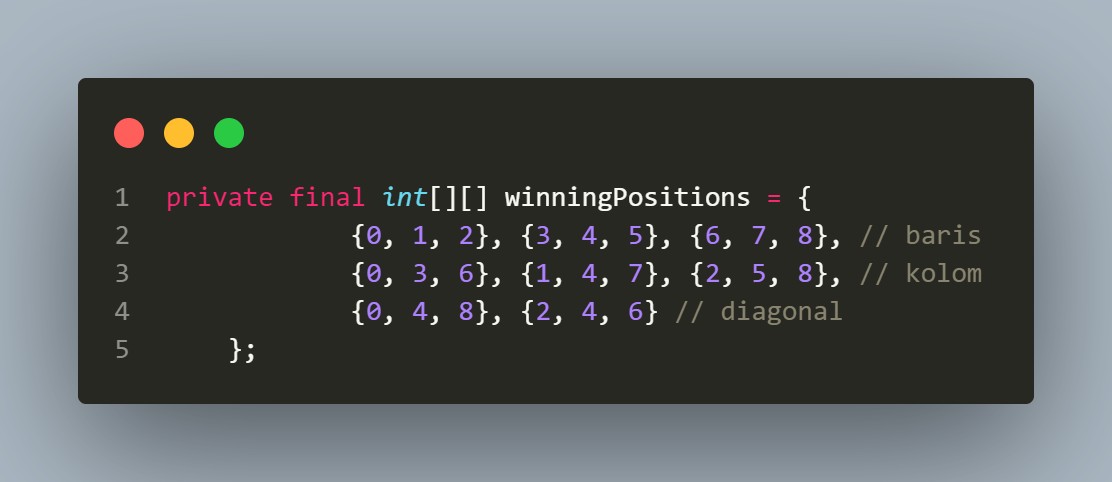
Penjelasan source code:

‘public static void main(String[] args) { ... }’ Ini adalah metode ‘main’, yang merupakan metode khusus dalam Java yang digunakan sebagai titik awal eksekusi program. Metode ini harus selalu memiliki tanda tangan yang sama (dengan argumen sebagai array ‘String[]’), sehingga JVM (Java Virtual Machine) tahu di mana harus memulai eksekusi program. ‘new TicTacToe();’ dalam metode ‘main’, sebuah objek dari kelas ‘TicTacToe’ dibuat menggunakan konstruktor tanpa argumen ‘(new TicTacToe();)’. Ini menciptakan sebuah instans dari kelas ‘TicTacToe’ yang memulai eksekusi program. Konstruktor ‘TicTacToe’ akan dijalankan saat objek baru dibuat, yang pada gilirannya akan memulai permainan ‘Tic-Tac-Toe’ sesuai dengan logika yang telah diimplementasikan dalam kelas ‘TicTacToe’.

* + Menyimpan Kombinasi Indeks Tombol

Printscreen

Kelompok 8



Gambar 1.11 Source code

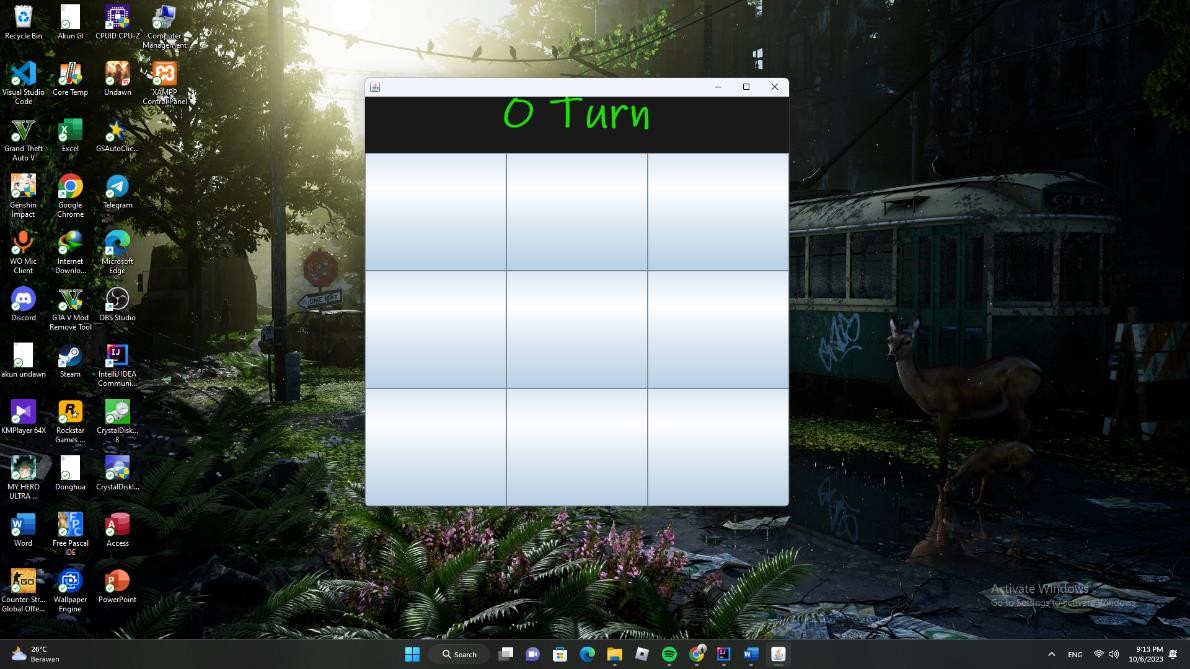
Penjelasan Source code:

‘private final int[][] winningPositions = { ... };’ Ini adalah deklarasi variabel array ‘winningPositions’ yang bersifat final (nilai tidak bisa diubah setelah diinisialisasi). Array ini menyimpan kombinasi indeks yang membentuk baris, kolom, dan diagonal yang bisa memenangkan permainan Tic-Tac-Toe. ‘{0, 1, 2}, {3, 4, 5}, {6, 7, 8}’ Ini adalah kombinasi indeks untuk baris-baris pada papan permainan Tic-Tac-Toe. Misalnya, baris pertama adalah tombol 0, 1, dan 2. ‘{0, 3, 6}, {1, 4, 7}, {2, 5, 8}’ Ini adalah kombinasi indeks untuk kolom- kolom pada papan permainan Tic-Tac-Toe. Misalnya, kolom pertama adalah tombol 0, 3, dan

1. ‘{0, 4, 8}, {2, 4, 6}’ Ini adalah kombinasi indeks untuk diagonal-diagonal pada papan permainan Tic-Tac-Toe. Diagonal pertama adalah tombol 0, 4, dan 8, sedangkan diagonal kedua adalah tombol 2, 4, dan 6.Output Papan
   * Output Papan

Printscreen

Kelompok 8



Gamabar 1.12 Screen output

Penjelasan Output code:

Saat ini, ini adalah tampilan Board yang dihasilkan oleh kode yang kita buat sebelumnya dengang menggunakan pemrograman java menggunakan beberapa library. Kita bisa memulai gamenya dengan mengklik kolom yang tersedia di atas. Karena yang duluan aladah player “O” maka saat kita klik nanti akan muncul gambar O jika bacaan di atas adalah X turn maka jika kita klik akan muncul gambar X.

* + Output X win

Printscreen

Kelompok 8



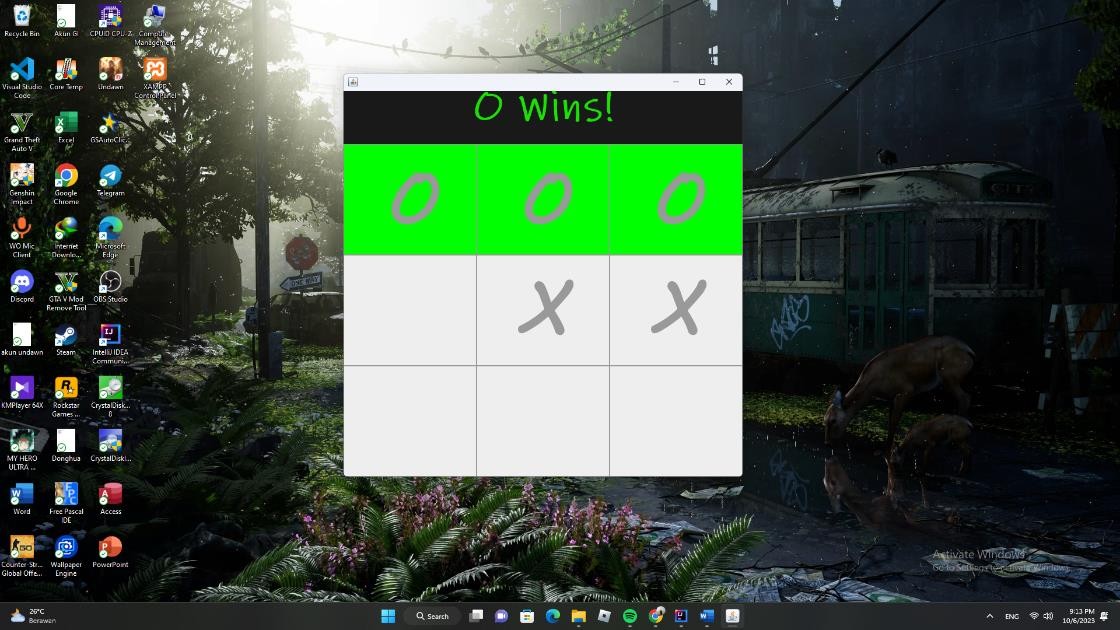
Gambar 1.13 Screen output

Penjelasan Output code:

Output tersebut menunjukkan bahwa pemain X telah menang dalam permainan Tic Tac Toe dengan menempatkan tanda X di semua kolom di baris ketiga.

* + Output 0 Menang

Printscreen



Kelompok 8

Gambar 1.14 screen output code

Penjelasan output code:

Pemain 0 Menang!" adalah pesan singkat yang menunjukkan 0 pemain yang menang game.

## KESIMPULAN DAN SARAN

1. **KESIMPULAN**

Bahasa pemrograman java berbasis objek, yang berarti hampir semua elemen dalam bahasa ini adalah objek. Ini memungkinkan pengembang untuk membuat kode yang mudah dimengerti, modular, dan dapat digunakan kembali. Salah satu fitur utama Java adalah kemampuannya untuk berjalan di berbagai platform (Platform Independence). Hal ini dicapai dengan menggunakan JVM (Java Virtual Machine), yang menerjemahkan kode Java menjadi bytecode yang dapat dijalankan di berbagai sistem operasi tanpa perlu melakukan perubahan pada kode sumber.

Software ini memiliki berbagai mekanisme keamanan seperti sandboxing dan pengendalian akses yang ketat yang membantu melindungi sistem dari ancaman keamanan. Kode Java dapat dengan mudah dipindahkan dari satu platform ke platform lainnya tanpa perlu melakukan banyak perubahan, Ini menjadikannya bahasa yang ideal untuk pengembangan perangkat lunak lintas platform.

Java menggunakan pengumpulan sampah (garbage collection) untuk mengelola memori. berarti pengembang tidak perlu secara eksplisit mengelola alokasi dan pembebasan memori. Java dilengkapi dengan libreri standar yang kaya yang mencakup berbagai fungsi dan kelas yang dapat digunakan pengembang untuk mempercepat pengembangan aplikasi. Juga memiliki dukungan bawaan untuk pemrograman berbasis multithreading, yang memungkinkan aplikasi untuk menjalankan beberapa tugas secara bersamaan. Ini berguna untuk meningkatkan kinerja dan responsifitas aplikasi.

## SARAN

Dengan memahami mata kuliah komputer dan pemrograman ini kita dapat mengetahui bagaimana bentuk kode pemrograman dari game tic tac toe.Saran saya jika kita belum sepenuhnya memahami materi dari mata kuliah komputer dan pemrograman kita perlu belajar mandiri agar lebih paham dan menambah wawasan kita.

Dalam project ini kita harus beradaptasi dengan kode kode pemrograman yang dipake, agar saat praktikum atau project selanjutnya kita sudah tidak bingung, serta dalam proses pembelajaran alangkah baiknya kita aktif bertanya dan memperhatikan tahapan-tahapan yang diajarkan, agar bisa mempermudah kita mengerjakan project atau tugas selanjutnya.

## DAFTAR PUSTAKA

*Buku Pertama Belajar Pemrograman JAVA*. (2013). Google Books.

https://books.google.co.id/books?hl=id&lr=&id=nijJEAAAQBAJ&oi=fnd&pg=PA3&d q=bahasa+pemrograman+java&ots=U9qaXRnO60&sig=dyLAaQ45rQpvi6TXIsXOvs8 GRFc&redir\_esc=y#v=onepage&q=bahasa%20pemrograman%20java&f=false

*Pemrograman Java Untuk Programmer*. (2015). Google Books. https://books.google.co.id/books?hl=id&lr=&id=6mh2DwAAQBAJ&oi=fnd&pg=PA3 &dq=pemrograman+java+&ots=B\_XhziMF3R&sig=wxG1AfaYolhARpvOeKYtZXb- FaY&redir\_esc=y#v=onepage&q=pemrograman%20java&f=false

*150 Rahasia Pemrograman Java*. (2023). Google Books. https://books.google.co.id/books?hl=id&lr=&id=jMQbTQp8rqwC&oi=fnd&pg=PR5& dq=+bahasa+pemrograman+java+&ots=ILbZIhHC79&sig=C- pzpT7MoH1aDJ76zR12keRS4HQ&redir\_esc=y#v=onepage&q=bahasa%20pemrogra man%20java&f=false