**UJIAN**

Mata Praktikum : Kecerdasan Buatan

Kelas : 3IA24

Praktikum ke- : 5

Tanggal : 12 Januari 2024

Materi : UJIAN

NPM : 50421704

Nama : Juan Samuel Christopher

Ketua Asisten : Brigitta Pricilia S

Nama Asisten :

Paraf Asisten :

Jumlah Lembar : 8



**LABORATORIUM TEKNIK INFORMATIKA**

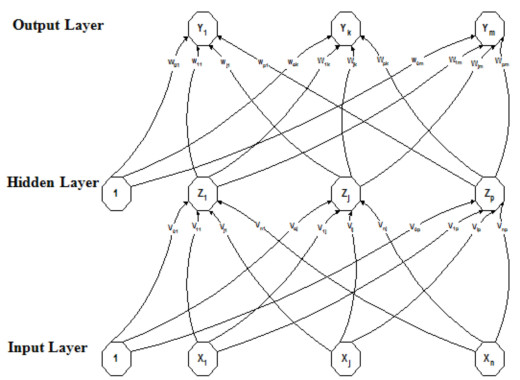
**UNIVERSITAS GUNADARMA**

**2024**

UJIAN KECERDASAN BUATAN

**SOAL UJIAN:**

1. Jelaskan gambar arsitektur dibawah ini



Jawab: Jaringan ini terstruktur dalam tiga lapisan. Pada lapisan input, tidak ada proses komputasi yang terjadi; namun, sinyal input X dikirimkan ke lapisan tersembunyi. Di lapisan tersembunyi dan lapisan output, terjadi proses komputasi yang melibatkan bobot dan bias, dengan perhitungan output dari kedua lapisan tersebut. Fungsi aktivasi yang digunakan dalam jaringan ini adalah sigmoid biner, dipilih karena menghasilkan output dalam rentang antara 0 hingga 1. Ini membantu dalam menghadirkan hasil yang sesuai dengan kebutuhan, di mana nilai output diharapkan berada dalam rentang tersebut. Dengan demikian, melalui tiga lapisan ini, jaringan dapat mengolah informasi dengan memperhitungkan bobot, bias, dan fungsi aktivasi sigmoid biner untuk menghasilkan output yang sesuai dengan karakteristik yang diinginkan.

1. jelaskan pengertian jaringan lapisan tunggal, jaringan dengan banyak lapisan dan jaringan lapisan kompetitif!

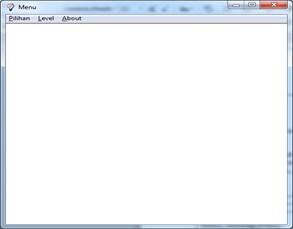
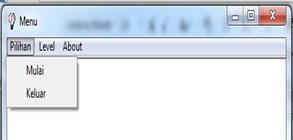
Jawab:

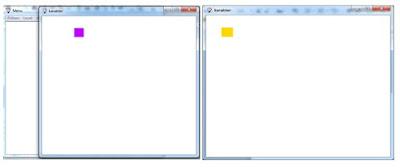
Jaringan lapisan tunggal adalah jenis arsitektur jaringan yang terdiri dari hanya satu lapisan, di mana setiap node atau neuron langsung terhubung dengan node atau neuron dalam lapisan berikutnya. Sebagai contoh, model perseptron adalah representasi sederhana dari jaringan lapisan tunggal, di mana input dikalikan dengan bobot tertentu dan hasilnya diproses melalui fungsi aktivasi.

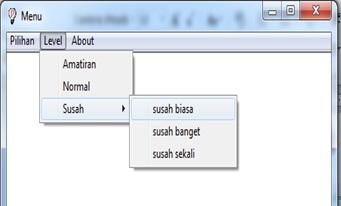
Kemuduan jaringan dengan banyak lapisan, atau yang lebih dikenal sebagai jaringan saraf tiruan yang mendalam, memiliki struktur yang lebih kompleks dengan beberapa lapisan tersembunyi antara lapisan input dan output. Keberadaan lapisan-lapisan ini memungkinkan jaringan untuk memahami representasi data yang lebih abstrak dan kompleks.

Terakhir, jaringan lapisan kompetitif adalah jenis jaringan saraf yang menekankan persaingan di antara neuron-neuron untuk menjadi aktif dalam merespons stimulus. Misalnya, dalam Self-Organizing Map (SOM), neuron-neuron bersaing untuk menjadi pemenang, yang membantu dalam tugas seperti klasifikasi atau pengelompokan data. Dengan demikian, ketiga jenis jaringan ini memberikan pendekatan yang berbeda dalam memproses informasi dan menangani berbagai tugas.

1. Buatlah program di strawberry prolog dengan output berikut serta berikan listing logika dan outputnya:

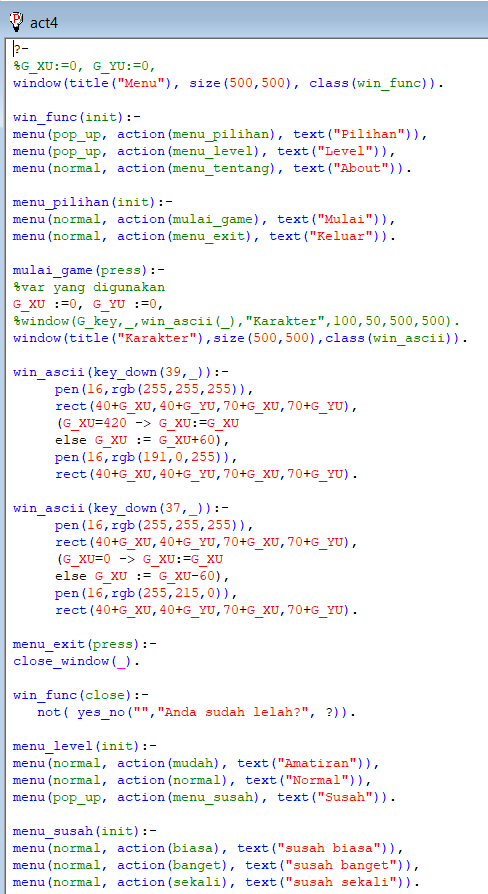
 

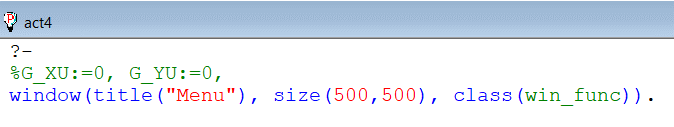


JAWAB:

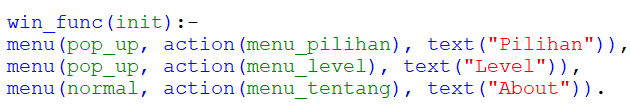
1. LISTING

****

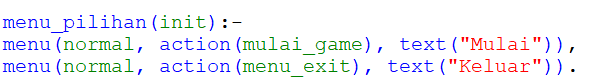
1. LOGIKA (SERTAKAN SS BARIS PROGRAM BARU BERIKAN PENJELASAN)



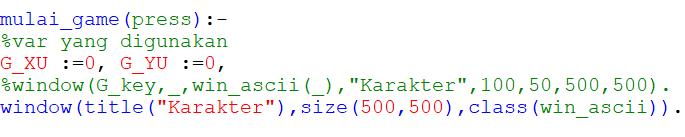
**?-** adalah prompt prolog yang menunggu request atau tujuan dari user untuk dijalankan. Variabel global %G\_XU dan G\_YU diinisialisasi dengan nilai 0, dan ini adalah sebuah komentar jadi tidak mempengaruhi eksekusi program. Kemudian, **window(title("Menu"), size(500,500), class(win\_func)**) berfungsi untuk membuat jendela dengan judul "Menu", ukuran 500x500 piksel, dan kelas "**win\_func**". Ini adalah bagian tampilan awal program yang menampilkan menu utama.



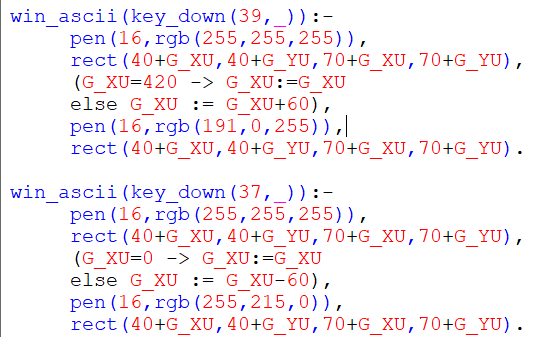
**win\_func(init):-** untuk menentukan aturan init dalam fungsi win\_func. Kemudian line selanjutnya mendefinisikan menu pop-up dengan aksi menu\_pilihan dan menu\_level, serta menu normal dengan aksi menu\_tentang. Ini mengatur opsi-opsi yang dapat dipilih dalam menu utama.



Pada blok program diatas **menu\_pilihan(init):-** berfungsi untuk menentukan aturan init dalam fungsi menu\_pilihan. Lalu **menu(normal, action(mulai\_game), text("Mulai")), menu(normal, action(menu\_exit), text("Keluar"))** berfungsi untuk mendefinisikan menu normal dengan aksi mulai\_game dan menu\_exit. Ini mengatur opsi-opsi yang dapat dipilih dalam menu pilihan.



Selanjutnya blok program diatas **mulai\_game(press):-** berfungsi untuk menentukan aturan press dalam fungsi mulai\_game. Ini mungkin berisi logika yang akan dijalankan saat pengguna memilih opsi "Mulai". Kemudian G\_XU := 0, G\_YU := 0, window(title("Karakter"),size(500,500),class(win\_ascii)) berfungsi untuk menginisialisasi variabel G\_XU dan G\_YU ke nilai 0, dan membuat jendela baru dengan judul "Karakter", ukuran 500x500 piksel, dan kelas "win\_ascii". Ini merupakan persiapan sebelum permainan dimulai.



Pada blok program diatas ini win\_ascii(key\_down(39,\_)):- ini adalah aturan dalam Prolog yang mendefinisikan aksi yang akan diambil ketika tombol panah kanan (kode 39) ditekan pada jendela dengan kelas "win\_ascii".

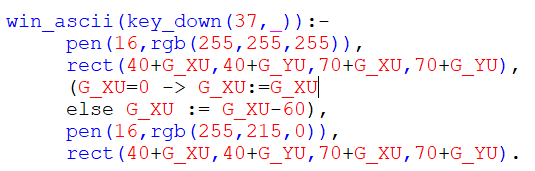
pen(16,rgb(255,255,255)), berfungsi untu mengatur pen (warna garis) dengan ketebalan 16 dan warna putih (RGB: 255, 255, 255).

rect(40+G\_XU,40+G\_YU,70+G\_XU,70+G\_YU), berfungsi untuk menggambar suatu persegi panjang (rectangle) pada posisi (40+G\_XU, 40+G\_YU) hingga (70+G\_XU, 70+G\_YU). Variabel G\_XU dan G\_YU digunakan untuk menentukan posisi karakter pada sumbu X dan Y.

(G\_XU=420 -> G\_XU:=G\_XU else G\_XU := G\_XU+60), ini adalah kondisional Prolog. Jika nilai G\_XU sudah mencapai 420, maka tidak ada perubahan nilai. Namun, jika tidak, maka nilai G\_XU akan ditambah 60.

pen(16,rgb(191,0,255)), berfungsi untuk mengatur pen dengan ketebalan 16 dan warna RGB (191, 0, 255).

rect(40+G\_XU,40+G\_YU,70+G\_XU,70+G\_YU). Berfungsi untuk menggambar persegi panjang baru pada posisi yang baru dihitung setelah adanya perubahan nilai G\_XU (jika kondisi di langkah 4 terpenuhi)



Pada blok program diatas **win\_ascii(key\_down(37,\_)):-**Ini adalah aturan dalam Prolog yang mendefinisikan aksi yang akan diambil ketika tombol panah kiri (kode 37) ditekan pada jendela dengan kelas "win\_ascii".

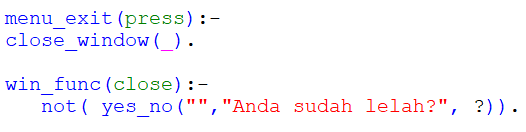
**pen(16,rgb(255,255,255)),** berfungsi untuk mengatur pen (warna garis) dengan ketebalan 16 dan warna putih (RGB: 255, 255, 255).

**rect(40+G\_XU,40+G\_YU,70+G\_XU,70+G\_YU)**, berfungsi untuk menggambar suatu persegi panjang (rectangle) pada posisi (40+G\_XU, 40+G\_YU) hingga (70+G\_XU, 70+G\_YU). Variabel G\_XU dan G\_YU digunakan untuk menentukan posisi karakter pada sumbu X dan Y.

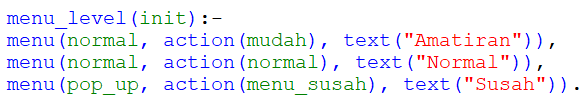
**(G\_XU=0 -> G\_XU:=G\_XU else G\_XU := G\_XU-60),**ini adalah kondisional Prolog. Jika nilai G\_XU sudah mencapai 0, maka tidak ada perubahan nilai. Namun, jika tidak, maka nilai G\_XU akan dikurangi 60.

**pen(16,rgb(255,215,0)),** berfungsi untukmengatur pen dengan ketebalan 16 dan warna RGB (255, 215, 0).

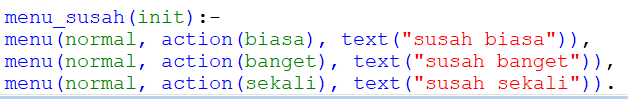
**rect(40+G\_XU,40+G\_YU,70+G\_XU,70+G\_YU).** Berfungsi untuk menggambar persegi panjang baru pada posisi yang baru dihitung setelah adanya perubahan nilai G\_XU (jika kondisi di langkah 4 terpenuhi).



Pada blok program diatas ini dijalankan saat tombol "Keluar" ditekan. Menutup jendela Fungsi ini dijalankan saat jendela ditutup. Menampilkan pesan konfirmasi "Anda sudah lelah?" dan menunggu jawaban yes/no



Pada blok program diatas ini menentukan menu pada inisialisasi "Level". Membuat menu normal "Amatiran" dan "Normal", dan pop-up menu "Susah"



Pada blok program diatas ini menentukan menu pada pop-up "Susah". Membuat menu normal "susah biasa", "susah banget", dan "susah sekali".

1. OUTPUT PROGRAM

