

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

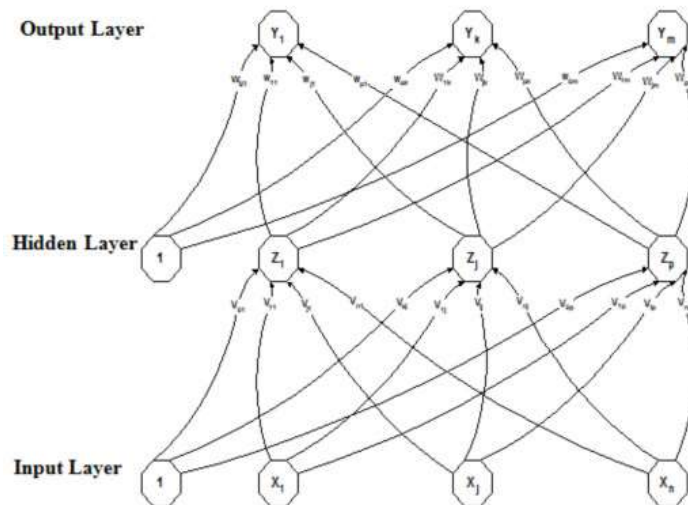
UNIVERSITAS GUNADARMA

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.

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

Jawab:

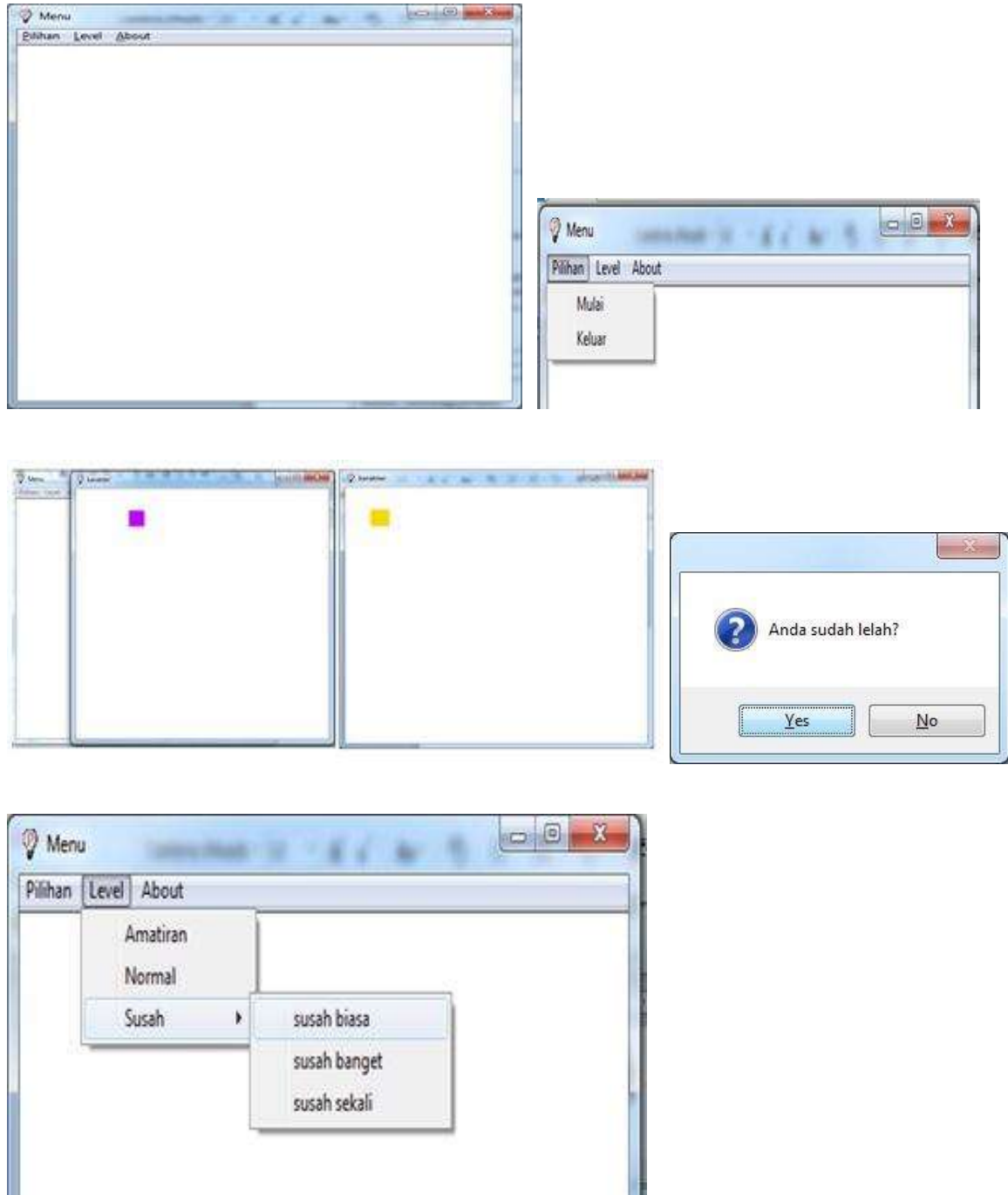
Jaringan lapisan tunggal adalah jenis arsitektur jaringan yang terdiri dari hanya satu lapisan, di mana setiap node atau neuron langsung terhubung dengan node atau neuron dalam lapisan berikutnya. Sebagai contoh, model perseptron adalah representasi sederhana dari jaringan lapisan tunggal, di mana input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

Kemudian 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.

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



JAWAB:

A. LISTING

 act4

```
|?-\n%G_XU:=0, G_YU:=0,\nwindow(title("Menu"), size(500,500), class(win_func)).\n\nwin_func(init):-\nmenu(pop_up, action(menu_pilihan), text("Pilihan")), \nmenu(pop_up, action(menu_level), text("Level")), \nmenu(normal, action(menu_tentang), text("About")).\n\nmenu_pilihan(init):-\nmenu(normal, action(mulai_game), text("Mulai")), \nmenu(normal, action(menu_exit), text("Keluar")).\n\nmulai_game(press):-\n%var yang digunakan\nG_XU :=0, G_YU :=0,\n%window(G_key,_,win_ascii(_),"Karakter",100,50,500,500).\nwindow(title("Karakter"),size(500,500),class(win_ascii)).\n\nwin_ascii(key_down(39,_)):-\n    pen(16,rgb(255,255,255)),\n    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU),\n    (G_XU=420 -> G_XU:=G_XU\n    else G_XU := G_XU+60),\n    pen(16,rgb(191,0,255)),\n    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU).\n\nwin_ascii(key_down(37,_)):-\n    pen(16,rgb(255,255,255)),\n    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU),\n    (G_XU=0 -> G_XU:=G_XU\n    else G_XU := G_XU-60),\n    pen(16,rgb(255,215,0)),\n    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU).\n\nmenu_exit(press):-\nclose_window(_).\n\nwin_func(close):-\n    not( yes_no("", "Anda sudah lelah?", ?)).\n\nmenu_level(init):-\nmenu(normal, action(mudah), text("Amatiran")), \nmenu(normal, action(normal), text("Normal")), \nmenu(pop_up, action(menu_susah), text("Susah")).\n\nmenu_susah(init):-\nmenu(normal, action(biasa), text("susah biasa")), \nmenu(normal, action(banget), text("susah banget")), \nmenu(normal, action(sekali), text("susah sekali")).
```

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

```
act4
?-
%G_XU:=0, G_YU:=0,
window(title("Menu"), size(500,500), class(win_func)).
```

?- 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):-
menu(pop_up, action(menu_pilihan), text("Pilihan")),
menu(pop_up, action(menu_level), text("Level")),
menu(normal, action(menu_tentang), text("About")).
```

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.

```
menu_pilihan(init):-
menu(normal, action(mulai_game), text("Mulai")),
menu(normal, action(menu_exit), text("Keluar")).
```

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.

```
mulai_game(press):-
%var yang digunakan
G_XU :=0, G_YU :=0,
%window(G_key,_,win_ascii(_), "Karakter",100,50,500,500).
window(title("Karakter"),size(500,500),class(win_ascii)).
```

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.

```
win_ascii(key_down(39,_)):-
    pen(16,rgb(255,255,255)),
    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU),
    (G_XU=420 -> G_XU:=G_XU
    else G_XU := G_XU+60),
    pen(16,rgb(191,0,255)),
    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU) .
```

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)

```
win_ascii(key_down(37,_)):-
    pen(16,rgb(255,255,255)),
    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU),
    (G_XU=0 -> G_XU:=G_XU
    else G_XU := G_XU-60),
    pen(16,rgb(255,215,0)),
    rect(40+G_XU,40+G_YU,70+G_XU,70+G_YU) .
```

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 untuk mengatur 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).

```
menu_exit(press):-  
    close_window(_).  
  
win_func(close):-  
    not( yes_no("", "Anda sudah lelah?", ?)).
```

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

```
menu_level(init):-  
    menu(normal, action(mudah), text("Amatiran")),  
    menu(normal, action(normal), text("Normal")),  
    menu(pop_up, action(menu_susah), text("Susah")).
```

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

```
menu_susah(init):-  
    menu(normal, action(biasa), text("susah biasa")),  
    menu(normal, action(banget), text("susah banget")),  
    menu(normal, action(sekali), text("susah sekali")).
```

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

C. OUTPUT PROGRAM

```
Compiling the file:  
C:\Users\juan samuel\Documents\LAB_TI_SEMESTER_5\KB\act4  
Warning 7: The predicate win_func(...) has an interrupted definition. (clause  
0 errors, 1 warning.  
  
Yes.
```

