Tugas Evaluasi Praktikum IV Makassar, 28 Oktober 2024

**PRAKTIKUM**

**STRUKTUR DATA**



Nama : Andi Ikhlas Mallomo

Stambuk : 13020230251

Frekuensi : TI\_SD-7 (A7)

Dosen : Syariful Mujaddid, S.Kom., M.T.

Asisten 1 : Annisa Pratama Putri

Asisten 2 : Rahma Puspitasari

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS MUSLIM INDONESIA**

**MAKASSAR**

**2024**

1. Jelaskan apa itu Single Linked List dan contoh penggunaannya pada sebuah aplikasi

Jawaban :

Single linked list adalah struktur data linier yang terdii dari rangkaian elemen yang diistilahkan sebagai *node* yang dimana setiap node menyimpan dua hal yaitu, data dan A computer chip with lines and dots

Description automatically generatedpointer ke node berikutnya. Hal ini yang membuatnya berbeda dengan array yang menggunakan indeks untuk mengakses elemen secara acaka, sedangkan single linked list secara berurutan.

Contoh penggunaan :

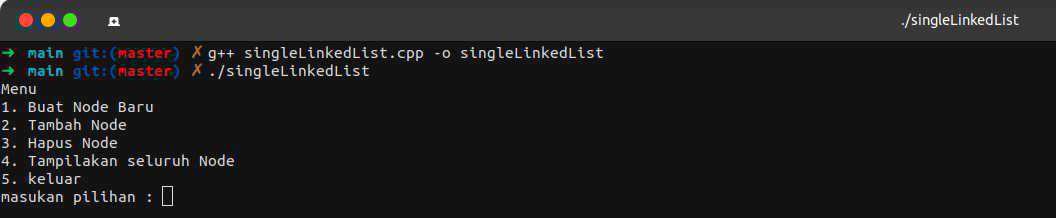
* Manajemen natrian di aplikasi bank atau klinik.

Single Linked List dapat digunakan untuk mengelola antrian pelanggan di bank atau pasien di klinik. Setiap pelanggan atau pasien yang baru datang dapat ditambahkan sebagai node baru di akhir daftar, dan pelanggan yang telah selesai dilayani dapat dihapus dari depan daftar.

* Navigasi dalam playlist musik

Dalam aplikasi pemutar musik, Single Linked List dapat digunakan untuk membuat playlist di mana setiap lagu dihubungkan ke lagu berikutnya. Ini mempermudah untuk berpindah dari satu lagu ke lagu berikutnya.

1. Buatlah implementasi dari Single Linked List dengan tahapan pengujian sebagai berikut:
2. Run program dan menghasilkan output



Jawaban :

CODE

#include <iostream>

using namespace std;

// Struktur untuk node dari linked list

struct Node {

int data; // Data yang disimpan di node

A computer chip with lines and dots

Description automatically generated Node\* next; // Pointer ke node berikutnya

};

Node\* head = nullptr; // Pointer ke head linked list

int lastDeletedValue = -1; // Variabel global untuk menyimpan nilai node terakhir yang dihapus

// Fungsi untuk membuat node baru di awal list

void buatNodeAwal(int nilai) {

head = new Node(); // Alokasikan memori untuk node baru

head->data = nilai; // Simpan nilai di node

head->next = nullptr; // Node baru menjadi node terakhir

cout << "Nilai " << nilai << " berhasil dimasukkan di node awal\n";

}

// Fungsi untuk menambahkan node di depan list

void tambahNodeDepan(int nilai) {

Node\* newNode = new Node(); // Alokasikan memori untuk node baru

newNode->data = nilai; // Simpan nilai di node

newNode->next = head; // Pointer ke node sebelumnya

head = newNode; // Update head ke node baru

cout << "Nilai " << nilai << " berhasil ditambahkan di depan\n";

}

// Fungsi untuk menambahkan node di belakang list

void tambahNodeBelakang(int nilai) {

Node\* newNode = new Node(); // Alokasikan memori untuk node baru

newNode->data = nilai; // Simpan nilai di node

newNode->next = nullptr; // Node baru menjadi node terakhir

A computer chip with lines and dots

Description automatically generated if (head == nullptr) { // Jika list kosong

head = newNode; // Head menjadi node baru

} else {

Node\* temp = head; // Temp pointer untuk traversing

while (temp->next != nullptr) { // Cari node terakhir

temp = temp->next;

}

temp->next = newNode; // Tambahkan node baru di belakang

}

cout << "Nilai " << nilai << " berhasil ditambahkan di belakang\n";

}

// Fungsi untuk menambahkan node setelah node dengan nilai tertentu

void tambahNodeSetelah(int nilaiBaru, int nilaiTarget) {

if (head == nullptr) { // Jika list kosong

cout << "List kosong, tidak dapat menambahkan node setelah nilai yang ditentukan.\n";

} else {

Node\* newNode = new Node(); // Alokasikan memori untuk node baru

newNode->data = nilaiBaru; // Simpan nilai baru

Node\* temp = head; // Temp pointer untuk traversing

while (temp != nullptr && temp->data != nilaiTarget) { // Cari nilai target

temp = temp->next;

}

if (temp == nullptr) { // Jika nilai target tidak ditemukan

cout << "Nilai " << nilaiTarget << " tidak ditemukan.\n";

} else {

A computer chip with lines and dots

Description automatically generated newNode->next = temp->next; // Update pointer node baru

temp->next = newNode; // Tempatkan node baru setelah nilai target

cout << "Nilai " << nilaiBaru << " berhasil ditambahkan setelah nilai " << nilaiTarget << ".\n";

}

}

}

// Fungsi untuk menghapus node di depan list

void hapusNodeDepan() {

if (head == nullptr) { // Jika list kosong

cout << "List kosong!\n";

return;

}

Node\* temp = head; // Simpan node yang akan dihapus

head = head->next; // Update head ke node berikutnya

lastDeletedValue = temp->data; // Simpan nilai node yang dihapus

cout << "Node pada posisi depan dengan nilai " << temp->data << " berhasil dihapus\n";

delete temp; // Hapus node

}

// Fungsi untuk menghapus node di belakang list

void hapusNodeBelakang() {

if (head == nullptr) { // Jika list kosong

cout << "List kosong!\n";

return;

}

A computer chip with lines and dots

Description automatically generated

if (head->next == nullptr) { // Jika hanya ada satu node

lastDeletedValue = head->data; // Simpan nilai node yang dihapus

cout << "Node pada posisi belakang dengan nilai " << head->data << " berhasil dihapus\n";

delete head; // Hapus node

head = nullptr; // Update head menjadi nullptr

} else {

Node\* temp = head; // Temp pointer untuk traversing

while (temp->next->next != nullptr) { // Cari node sebelum terakhir

temp = temp->next;

}

lastDeletedValue = temp->next->data; // Simpan nilai node yang dihapus

cout << "Node pada posisi belakang dengan nilai " << temp->next->data << " berhasil dihapus\n";

delete temp->next; // Hapus node terakhir

temp->next = nullptr; // Update pointer node sebelumnya

}

}

// Fungsi untuk menghapus node dengan nilai tertentu

void hapusNodeTengah(int nilai) {

if (head == nullptr) { // Jika list kosong

cout << "List kosong!\n";

return;

}

if (head->data == nilai) { // Jika node pertama yang dihapus

A computer chip with lines and dots

Description automatically generated hapusNodeDepan();

return;

}

Node\* temp = head; // Temp pointer untuk traversing

while (temp->next != nullptr && temp->next->data != nilai) { // Cari node yang ingin dihapus

temp = temp->next;

}

if (temp->next == nullptr) { // Jika nilai tidak ditemukan

cout << "Node dengan nilai " << nilai << " tidak ditemukan!!\n";

} else {

Node\* delNode = temp->next; // Simpan node yang akan dihapus

temp->next = delNode->next; // Update pointer

cout << "Node dengan nilai " << nilai << " berhasil dihapus\n";

delete delNode; // Hapus node

}

}

// Fungsi untuk menampilkan semua data dalam linked list

void tampilData() {

if (head == nullptr) { // Jika list kosong

cout << "List kosong!\n";

return;

}

Node\* temp = head; // Temp pointer untuk traversing

while (temp != nullptr) { // Tampilkan setiap node

A computer chip with lines and dots

Description automatically generated cout << temp->data << " -> ";

temp = temp->next;

}

cout << "NULL\n"; // Tampilkan akhir list

// Tampilkan nilai node terakhir yang dihapus jika ada

if (lastDeletedValue != -1) {

cout << "Node terakhir yang dihapus: " << lastDeletedValue << "\n";

lastDeletedValue = -1; // Reset setelah ditampilkan

}

}

// Menu untuk menambah node

void menuTambahNode() {

int pilihan, nilai, nilaiTarget;

if (head == nullptr) {

cout << "Node Belum Tersedia\n"; // Jika tidak ada node

} else {

do {

system("clear"); // Bersihkan layar

cout << "Menu Tambah Node\n================\n";

cout << "1. Tambah Node di Depan\n2. Tambah Node di Belakang\n3. Tambah Node Di Tengah\n4. Batal\n";

cout << "Masukkan pilihan: ";

cin >> pilihan;

switch (pilihan) {

case 1: // Tambah node di depan

cout << "Masukkan nilai: ";

cin >> nilai;

A computer chip with lines and dots

Description automatically generated if (cin.fail()) { // Cek input

cin.clear(); // Bersihkan flag kesalahan

cin.ignore(numeric\_limits<streamsize>::max(), '\n'); // Abaikan input yang tidak valid

cout << "Input is not valid.\n"; // Tampilkan pesan kesalahan

} else {

tambahNodeDepan(nilai); // Panggil fungsi

}

break;

case 2: // Tambah node di belakang

cout << "Masukkan nilai: ";

cin >> nilai;

if (cin.fail()) { // Cek input

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Input is not valid.\n"; // Tampilkan pesan kesalahan

} else {

tambahNodeBelakang(nilai); // Panggil fungsi

}

break;

case 3: // Tambah node setelah nilai tertentu

cout << "Masukkan nilai: ";

cin >> nilai;

if (cin.fail()) { // Cek input

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Input is not valid.\n"; // Tampilkan pesan kesalahan

} else {

tampilData(); // Tampilkan data sebelum menambah

cout << "Masukkan nilai setelah nilai: ";

cin >> nilaiTarget;

if (cin.fail()) { // Cek input

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Input is not valid.\n"; // Tampilkan pesan kesalahan

} else {

tambahNodeSetelah(nilai, nilaiTarget); // Panggil fungsi

}

A computer chip with lines and dots

Description automatically generated }

break;

case 4: // Batal

cout << "Batal.\n";

break;

default: // Pilihan tidak valid

cout << "Pilihan tidak valid!\n";

}

system("read -p 'Press Enter to continue...' var"); // Tunggu input

} while (pilihan != 4); // Ulangi hingga batal

}

}

// Menu untuk menghapus node

void menuHapusNode() {

int pilihan, nilai;

do {

system("clear"); // Bersihkan layar

cout << "Menu Hapus Node\n===============\n";

cout << "1. Hapus Node di Depan\n2. Hapus Node di Belakang\n3. Hapus Node di Tengah\n4. Batal\n";

cout << "Masukkan pilihan: ";

cin >> pilihan;

switch (pilihan) {

case 1: // Hapus node di depan

A computer chip with lines and dots

Description automatically generated hapusNodeDepan();

break;

case 2: // Hapus node di belakang

hapusNodeBelakang();

break;

case 3: // Hapus node di tengah

cout << "Masukkan nilai yang akan dihapus: ";

cin >> nilai;

hapusNodeTengah(nilai); // Panggil fungsi

break;

case 4: // Batal

break;

default: // Pilihan tidak valid

cout << "Pilihan tidak valid!\n";

}

system("read -p 'Press Enter to continue...' var"); // Tunggu input

} while (pilihan != 4); // Ulangi hingga batal

}

int main() {

int pilihan, nilai;

A computer chip with lines and dots

Description automatically generated

do {

system("clear"); // Bersihkan layar

cout << "Menu:\n1. Buat Node Baru\n2. Tambah Node\n3. Hapus Node\n4. Tampilkan Seluruh Node\n5. Keluar\n";

cout << "Masukkan pilihan: ";

cin >> pilihan;

switch (pilihan) {

case 1: // Buat node baru

if (head == nullptr) {

cout << "Buat Node Baru\n==============\n";

cout << "Masukkan nilai: ";

cin >> nilai;

buatNodeAwal(nilai); // Panggil fungsi

} else {

cout << "Node awal sudah dibuat!\n"; // Jika sudah ada node

}

break;

case 2: // Menu tambah node

menuTambahNode();

break;

case 3: // Menu hapus node

menuHapusNode();

break;

case 4: // Tampilkan seluruh node

tampilData();

break;

case 5: // Keluar dari program

cout << "Keluar dari program.\n";

A computer chip with lines and dots

Description automatically generated break;

default: // Pilihan tidak valid

cout << "Pilihan tidak valid!\n";

}

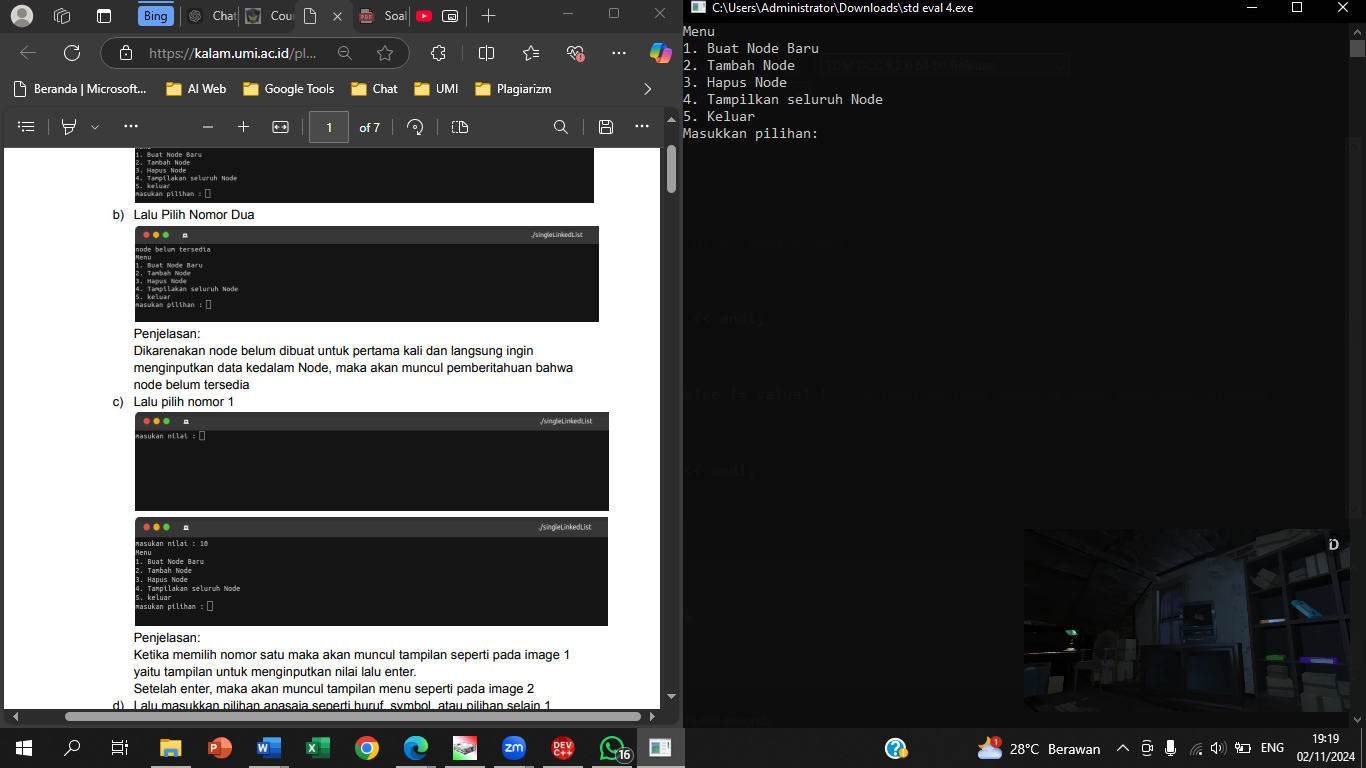
system("read -p 'Press Enter to continue...' var"); // Tunggu input

} while (pilihan != 5); // Ulangi hingga keluar

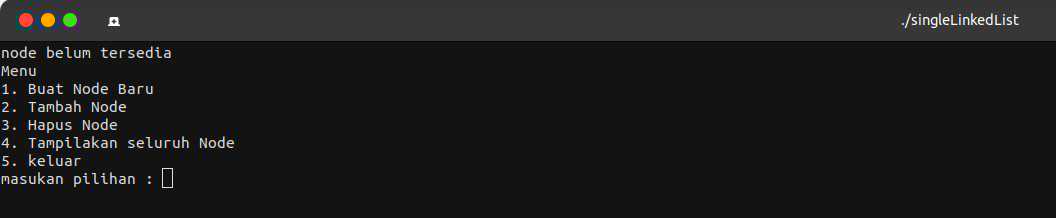
return 0; // Akhir program

}

OUTPUT



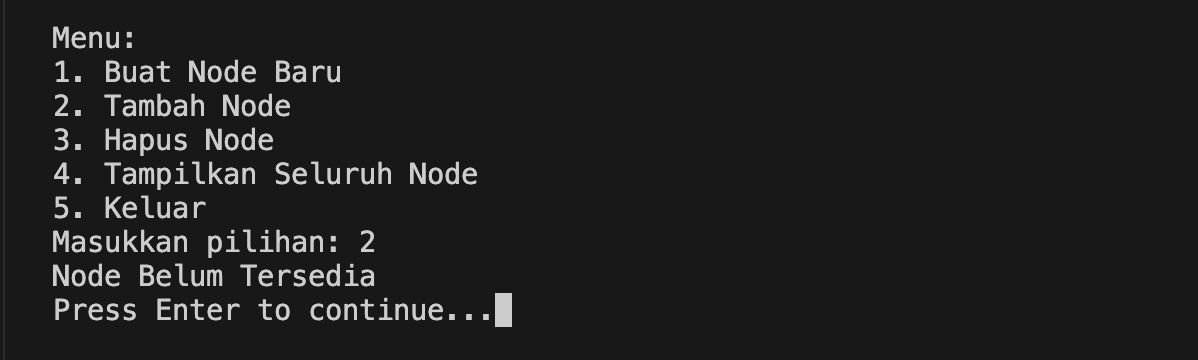
1. Lalu pilih nomor dua



Penjelasan :

Dikarenakan node belum dibuat untuk pertama kali dan langsung ingin menginputkan data kedalam Node, maka akan muncul pemberitahuan bahwa node belum tersedia

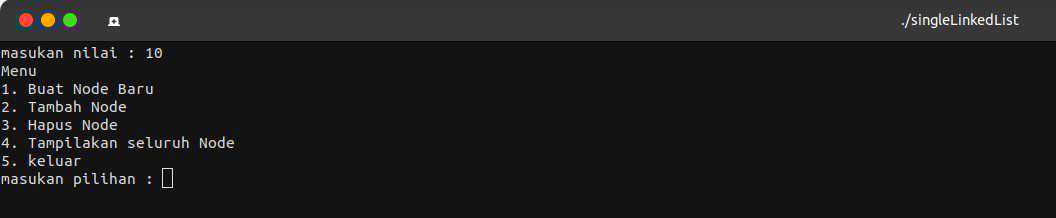
Jawaban:



1. A computer chip with lines and dots

   Description automatically generatedLalupilihnomor1A black rectangular object with a black border

   Description automatically generated

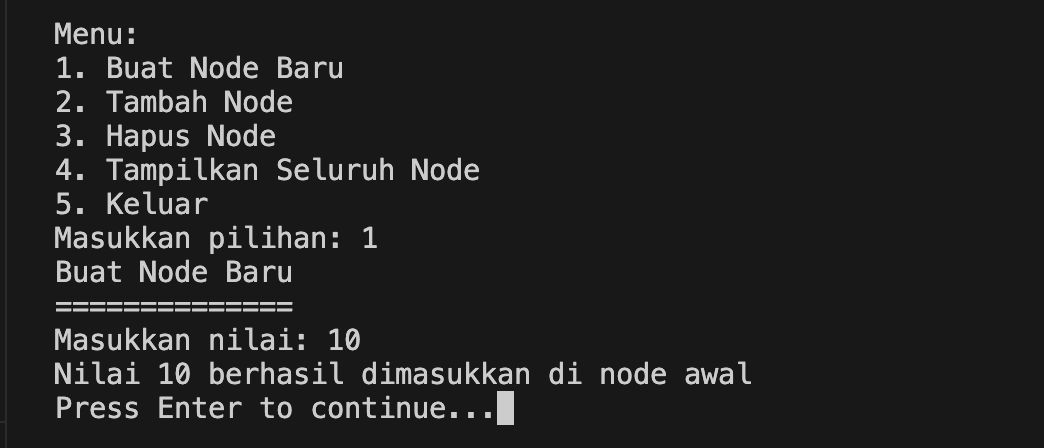


Penjelasan :

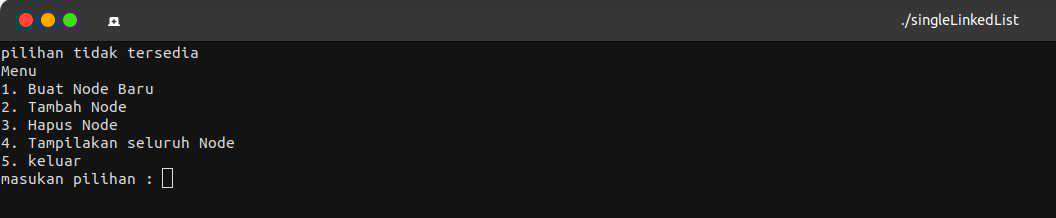
Ketika memilih nomor satu maka akan muncul tampilan seperti pada A computer chip with lines and dots

Description automatically generatedimage 1 yaitu tampilan untuk menginputkan nilai lalu enter. Setelah enter, maka akan muncul tampilan menu seperti pada image 2

Jawaban :



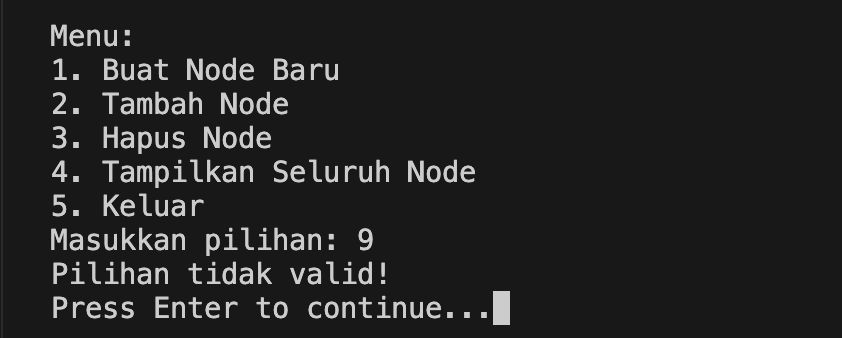
1. Lalu masukkan pilihan apa saja seperti huruf simbol atau piihan selain 1 sampai dengan 5



Penjelasan :

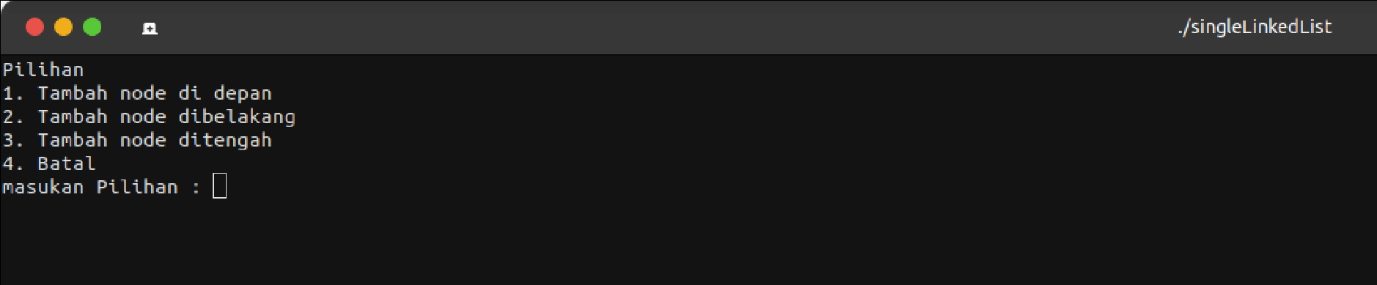
Ketika memilih selain dari pilihan pada menu (symbol, angka, ataupun huruf), maka akan memunculkan pemberitahuan seperti pada image 1

Jawaban :



1. Lalu pilih nomor 2 untuk memasukkan data pada node

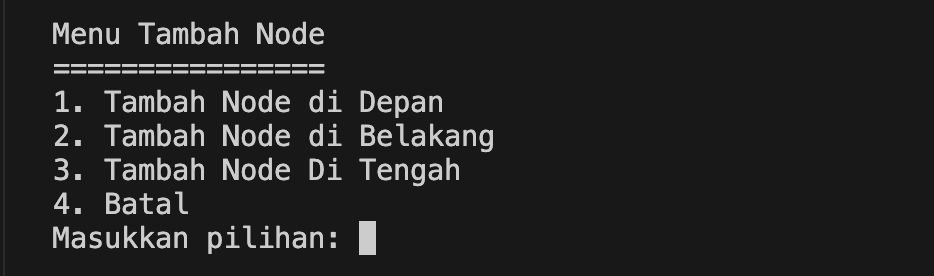
A computer chip with lines and dots

Description automatically generated

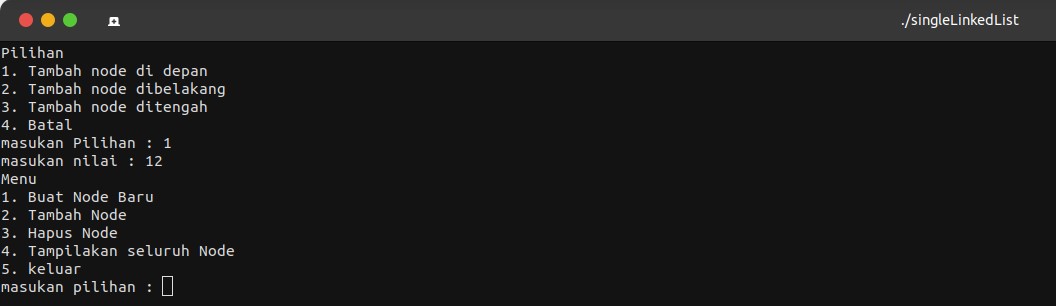
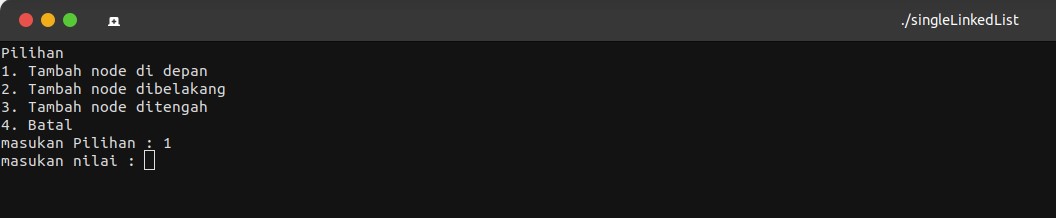
Penjelasan :

Akan muncul output seperti diatas yang dimana diminta untuk memilih menambahkan node di depan, tengah atau belakang

Jawaban :



1. Lalu pilih nomor 1 untuk menambahkan Node di depan

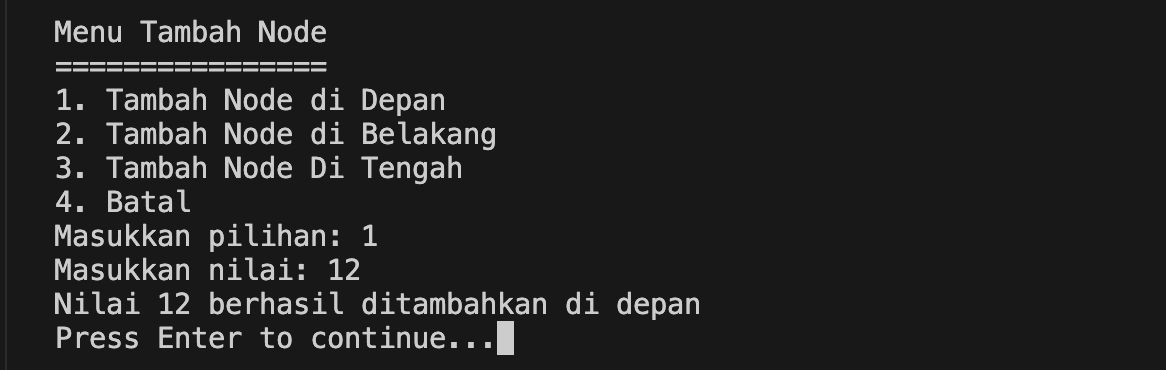


A computer chip with lines and dots

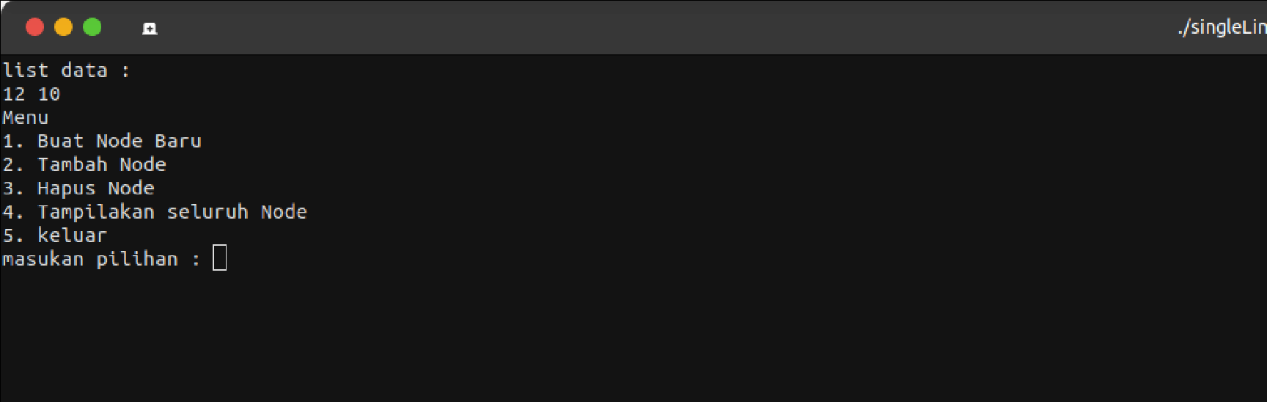
Description automatically generatedPenjelasan :

ketika kita enter pada pilihan 1, maka akan muncul output untuk menginputkan nilai pada Node seperti yang terlihat pada image 1 setelah menginputkan nilai lalu enter, maka akan kembali pada menu utama seperti yang terlihat pada image 2

Jawabam :



1. Lalu pilih nomor 4 untuk menampilkan data yang dimasukkan sebelumnya

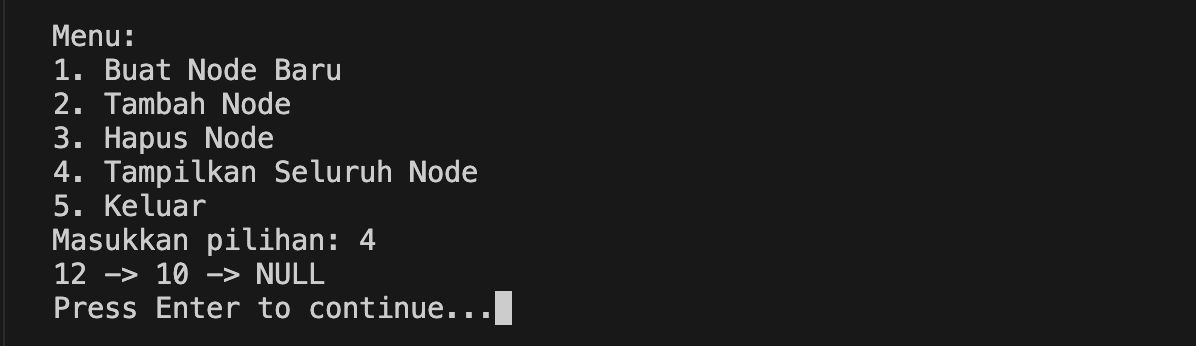


Penjelasan :

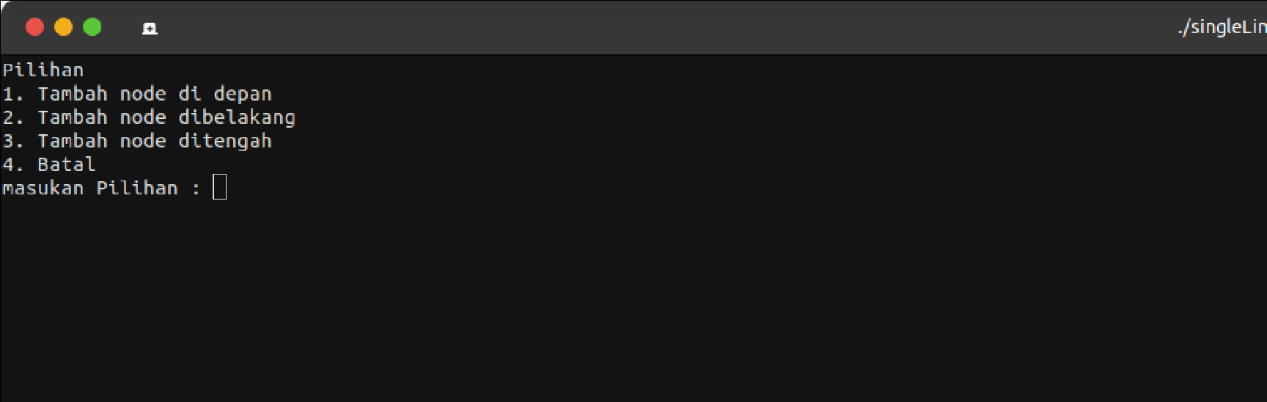
A computer chip with lines and dots

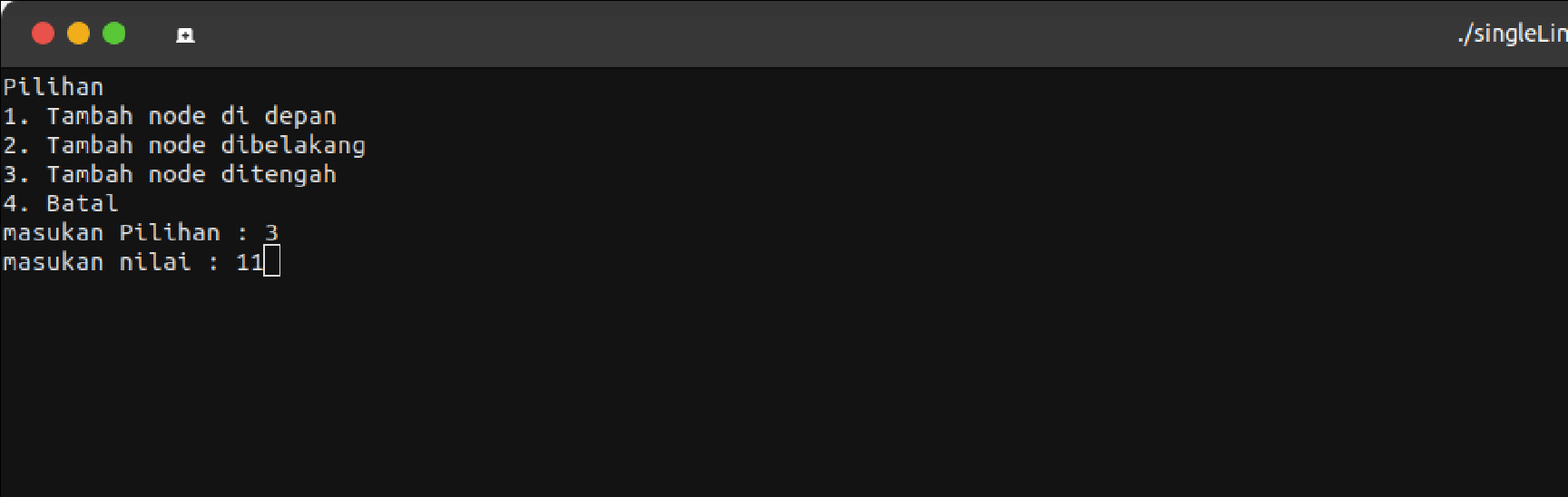
Description automatically generatedKetika kita memilih menu 4, maka akan tampil data yang dimasukkan sebelumnya. Lihat ada dua data yang tampil padahal proses memasukkan data baru satu kali. Data tersebut muncul dari ketika kita membuat Node untuk pertama kali. Data node yang dimasukkan untuk pertama kali akan langsung tersimpan pada Node.

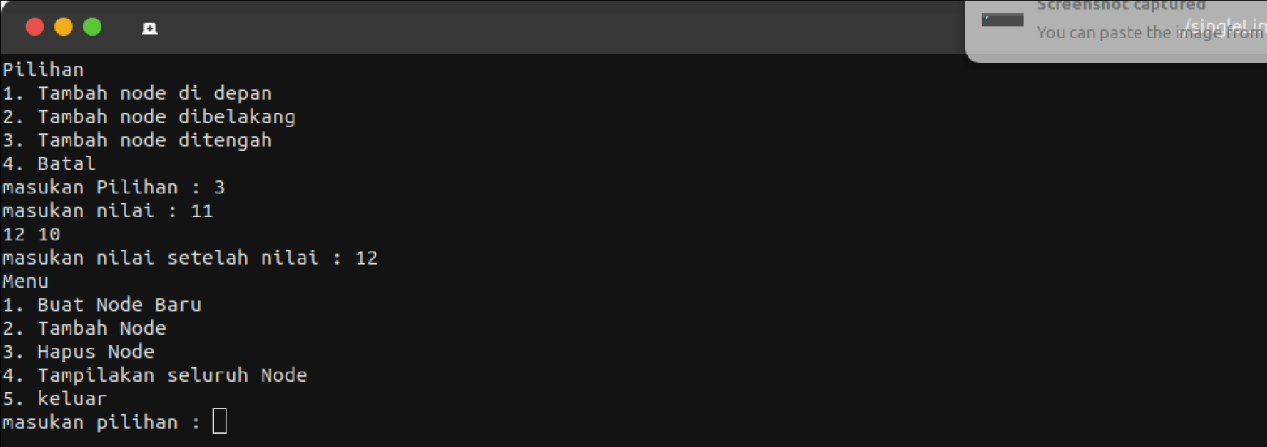
Jawaban :



1. Lalu pilih kembali menu 2 untuk memasukkan data





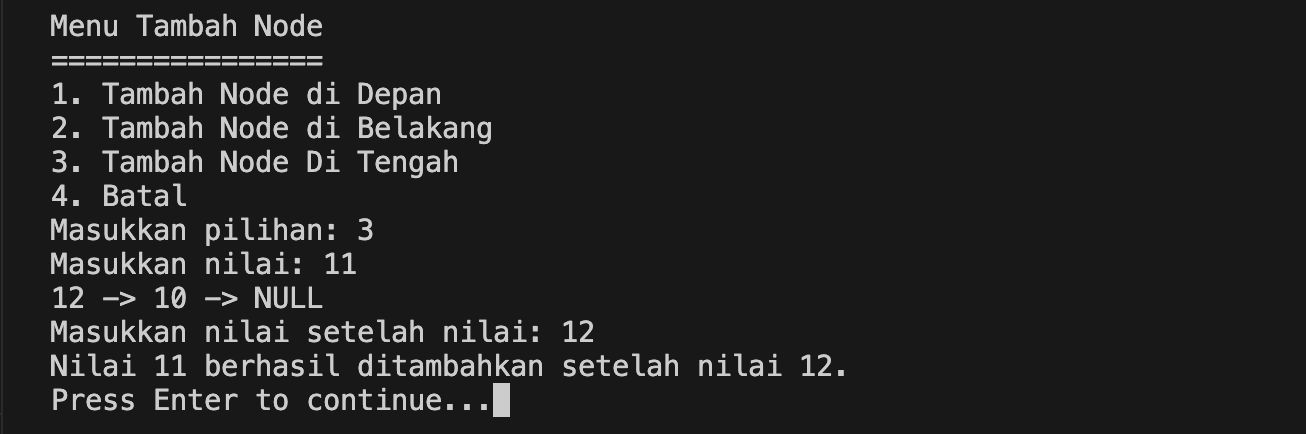


Penjelasan : A computer chip with lines and dots

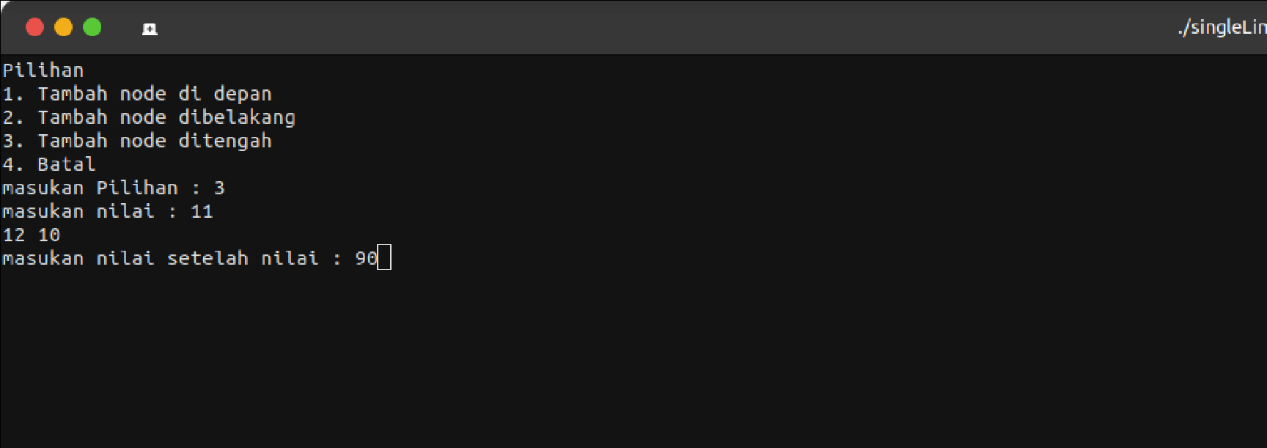
Description automatically generated

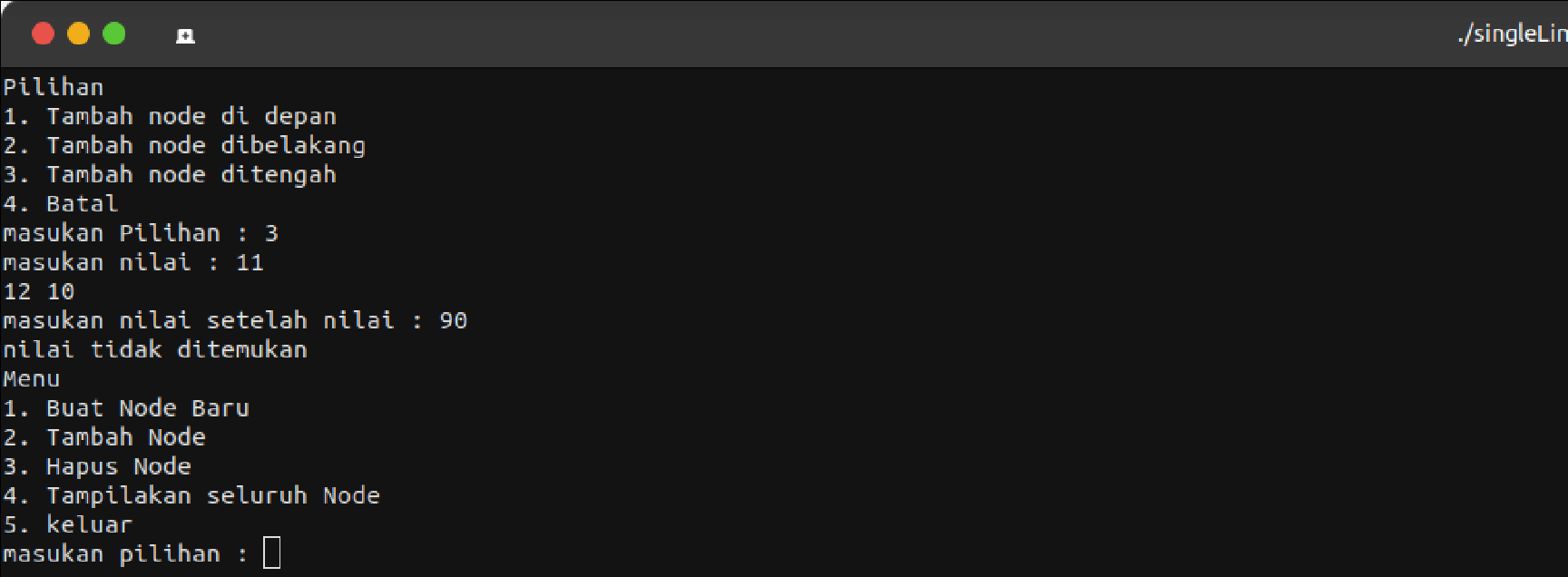
Pada percobaan untuk penginputan data untuk yang kedua kalinya, pilih menu 3 untuk menginputkan data di tengah seperti pada image 1 dan 2 lalu ketika nilai telah dimasukkan lalu enter, maka akan muncul tampilan list data yang ada di dalam Node, kemudian sistem akan meminta untuk data yang akan dimasukkan akan di tersimpan setelah data keberapa seperti pada image 3.

Jawaban :



1. Lalu ulangi tahapan pada nomor 3



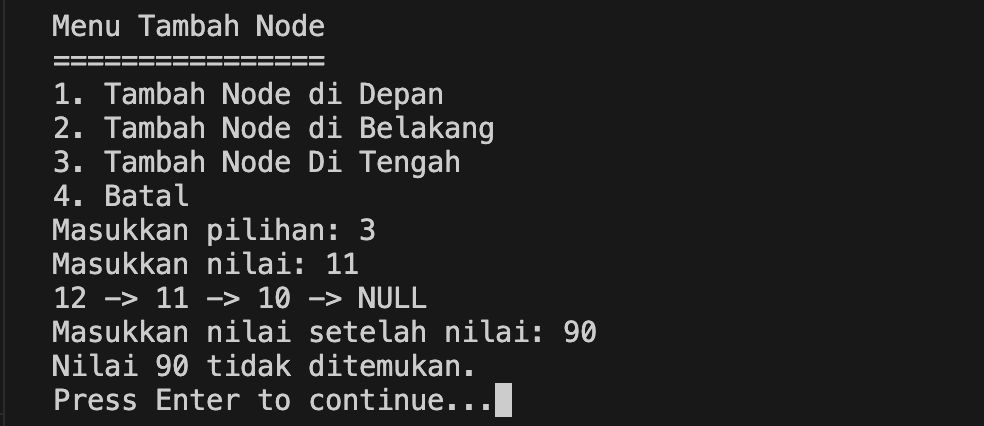


Penjelasan :

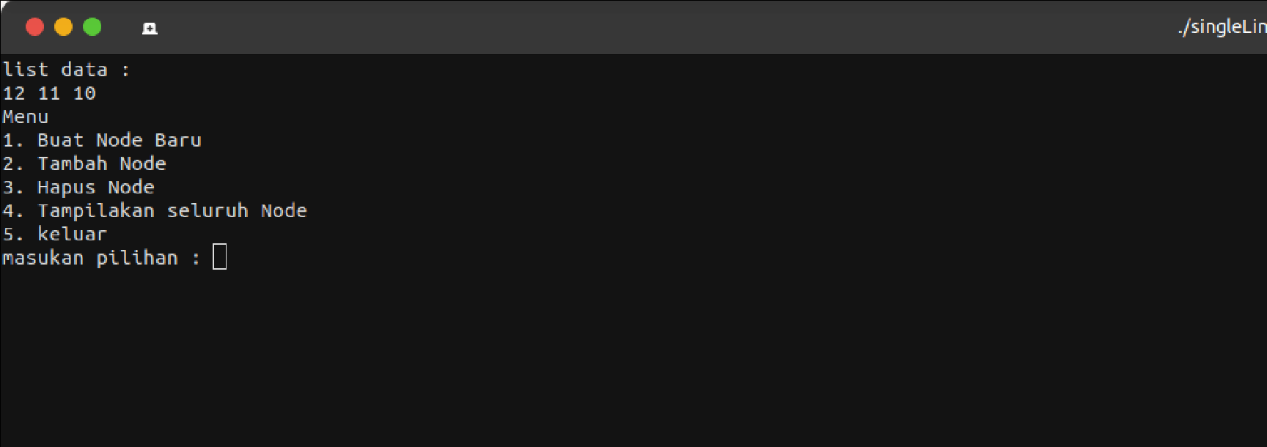
A computer chip with lines and dots

Description automatically generatedPada tahapan ini, menu yang dipilih adalah tambah data Node di tengah. Namun, setelah diminta untuk memasukkan data dan diminta untuk memasukan data setelah data Node ke berapa ? dan cobalah untuk memasukkan data yang tidak tersedia dan mengeluarkan pemberitahuan “data tidak ditemukan” dan penginputan yang dilakukan gagal. Contoh output dapat dilihat pada image 1 dan 2

Jawaban :



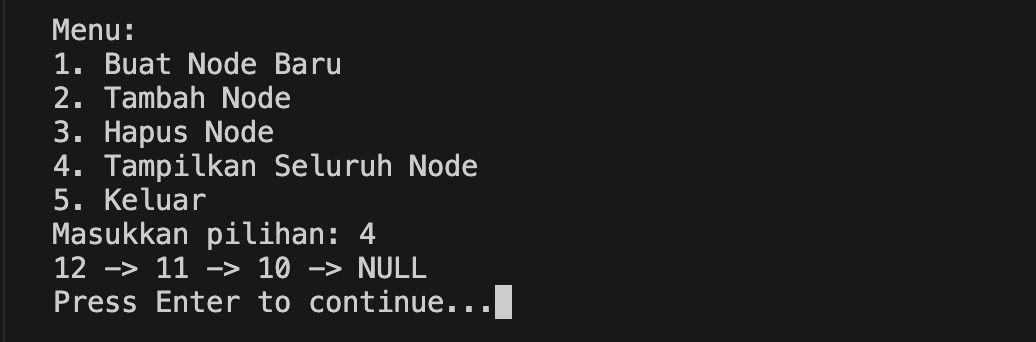
1. Lalu pilih kembali menu 4 untuk menampilkan list Node



Penjelasan :

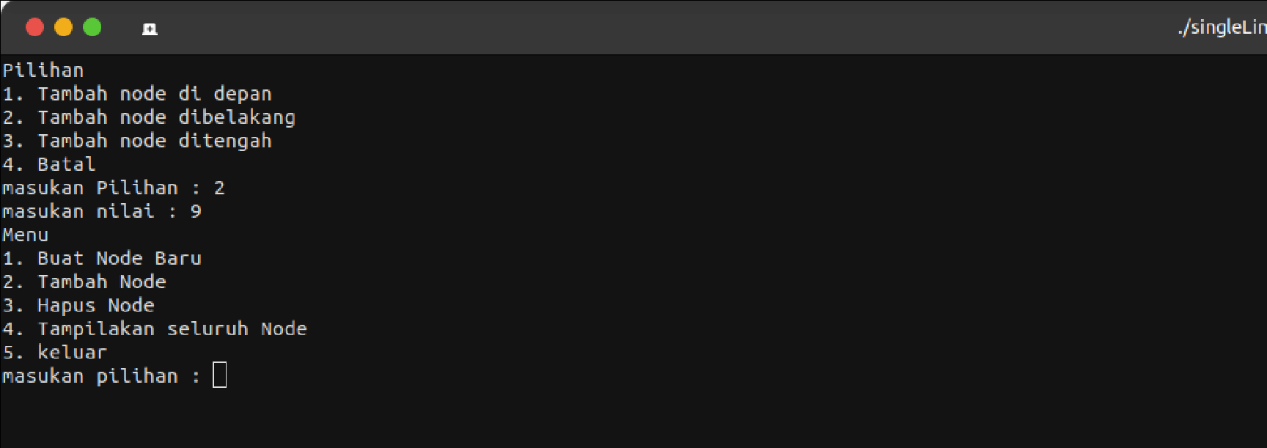
Seperti yang terlihat pada image 1, data yang diinputkan sebelumnya dengan menu input Node di tengah dan memilih untuk disimpan setelah data 12, maka data tersebut akan disimpan di tengah dan setelah data 12

Jawaban :



1. A computer chip with lines and dots

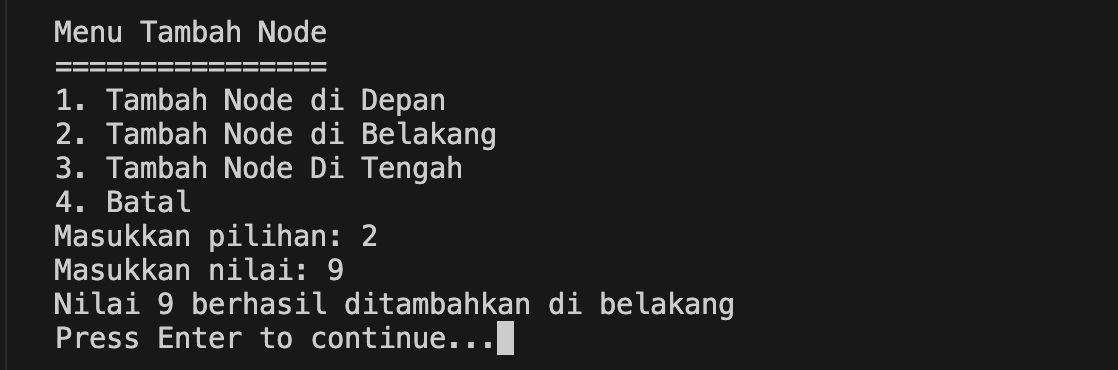
   Description automatically generatedLalu pilih kembali menu 2 untuk memasukkan node



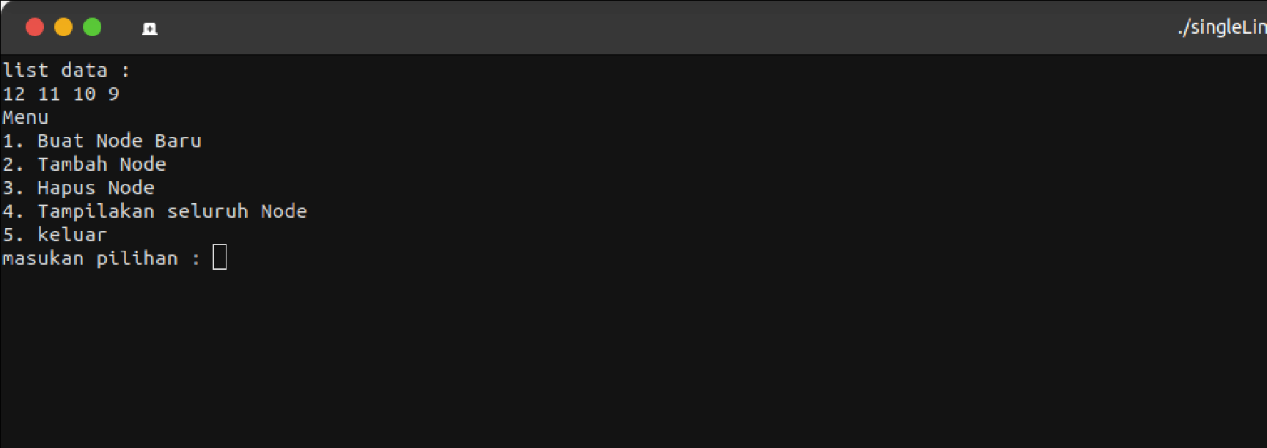
Penjelasan :

Pada tahap pengujian ini, pilih menu nomor 2 untuk menginputkan data Node di belakang seperti yang terlihat pada image 1

Jawaban :



1. Lalu pilih kembali menu 4 untuk menampilkan data yang dimasukkan sebelumnya di bagian belakang

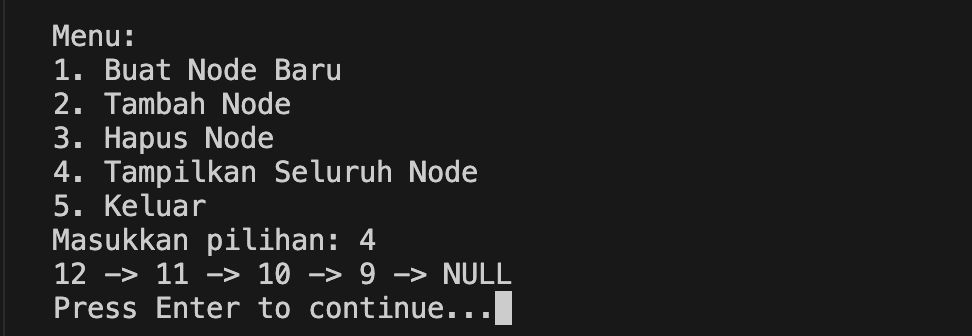


A computer chip with lines and dots

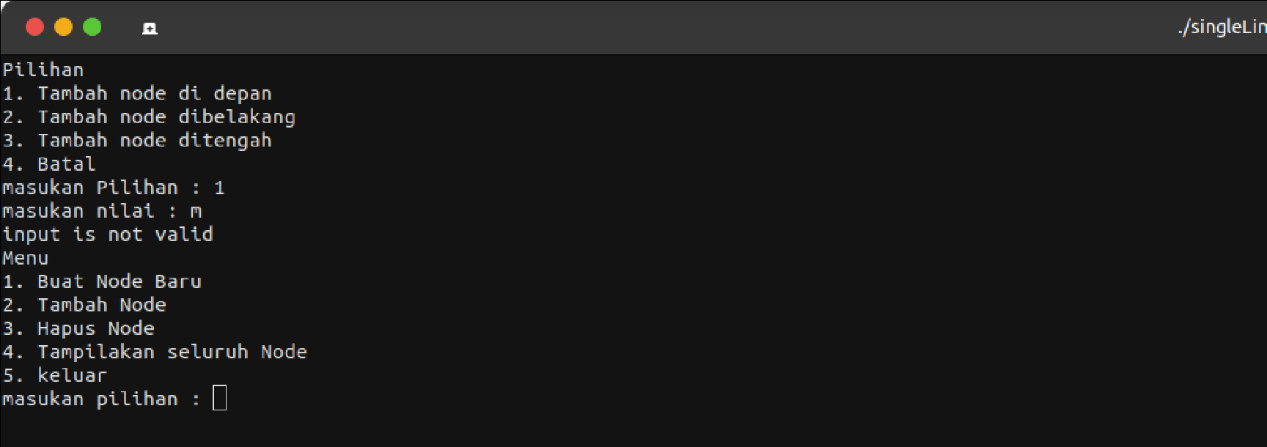
Description automatically generatedPenjelasan :

Seperti yang terlihat tahapan ini, data yang kita inputkan akan masuk pada bagian belakang seperti yang terlihat pada image 1

Jawaban :



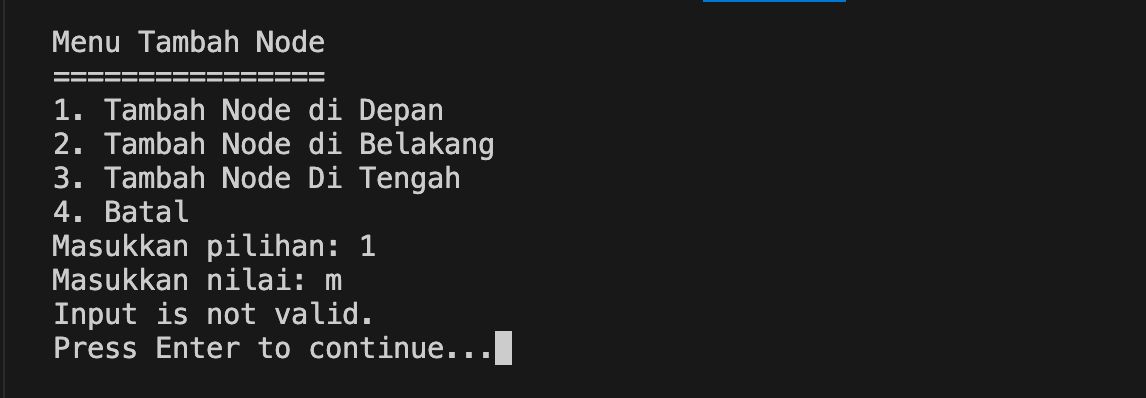
1. Lalu pilih kembali menu 2 untuk memasukkan data



Penjelasan :

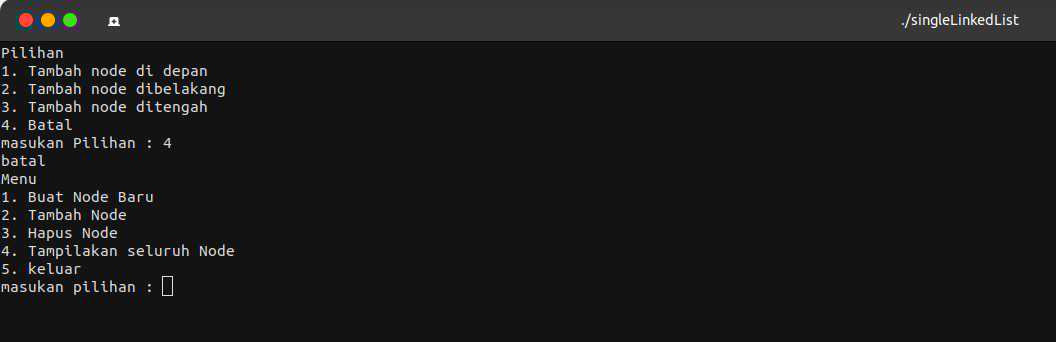
Pada tahapan ini, cobalah untuk menginputkan data seperti huruf dan simbol, maka akan muncul pemberitahuan bahwa “input is not valid” seperti yang terlihat pada image 1, Lakukan percobaan ini beberapa kali dan tampilkan outputnya

Jawaban :



1. LA computer chip with lines and dots

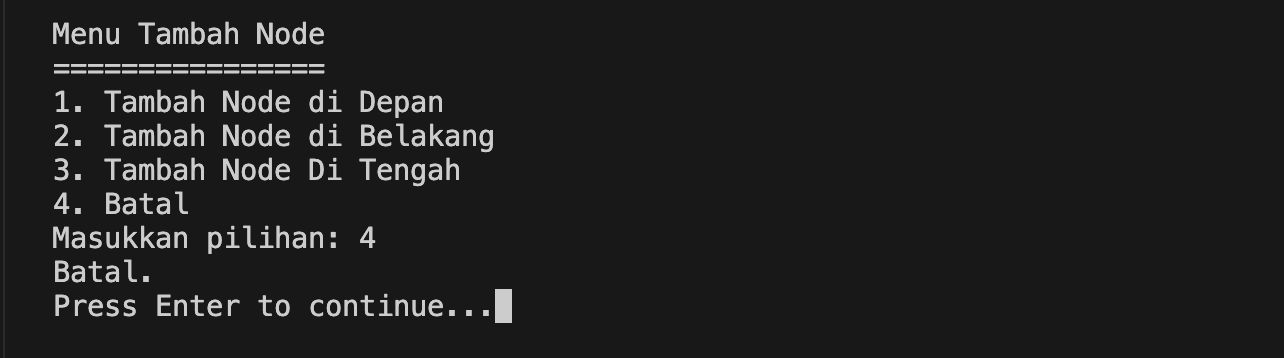
   Description automatically generatedalu pilih kembali menu nomor 2



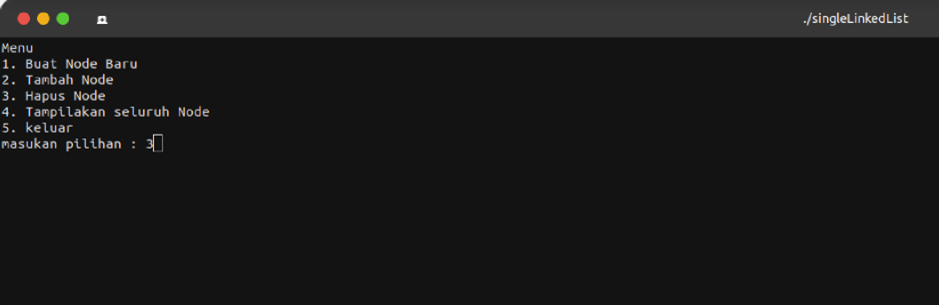
Penjelasan :

pada tahapan ini, cobalah untuk membatalkan proses inputan nilai, maka akan kembali pada menu utama dibarengi dengan pemberitahuan “batal” seperti yang terlihat pada image 1

Jawaban :

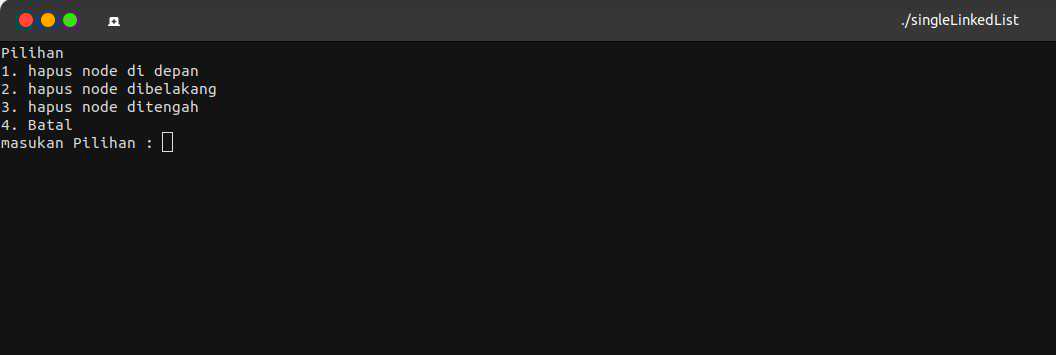


1. setelah proses sebelumnya berhasil semua, maka dilanjutkan dengan memilih menu 3 untuk menghapus node

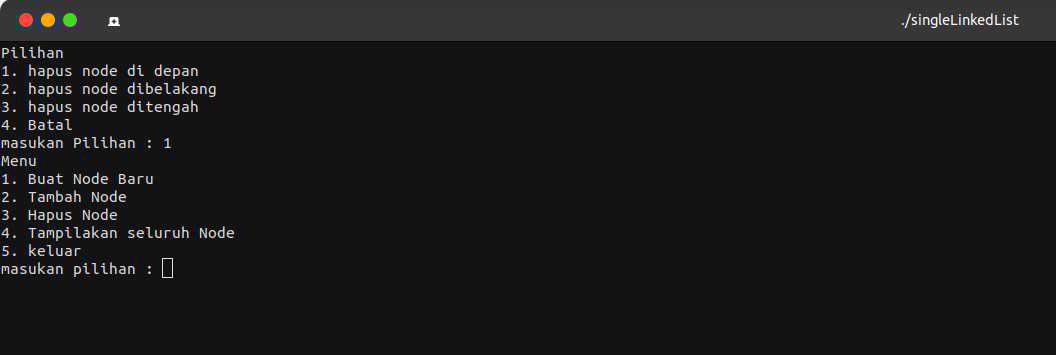


1. A computer chip with lines and dots

   Description automatically generatedSetelah memilih menu 3 maka akan muncul pilihan seperti gambar dibawah



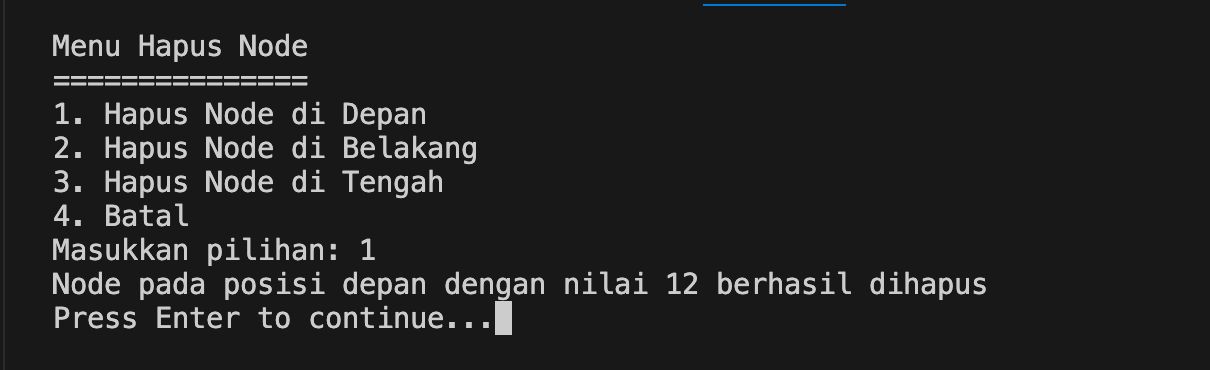
1. Lalu pilih nomor 1 untuk menghapus data Node di depan



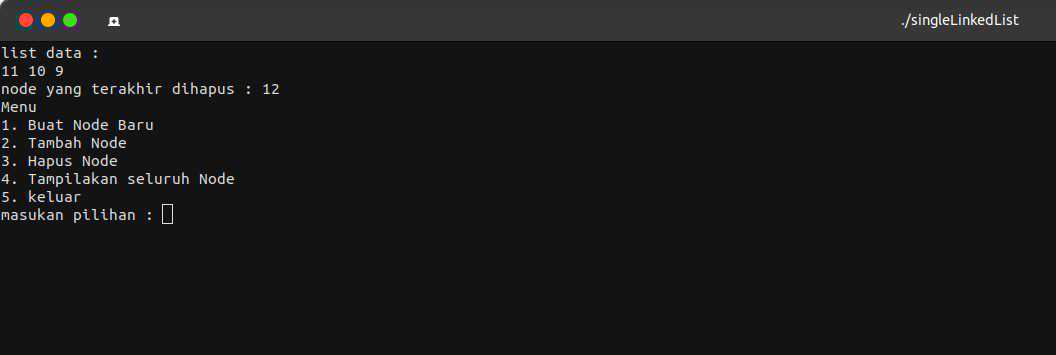
Penjelasan :

Setelah memilih menu 1 pada menu hapus Node, maka akan kembali ke menu utama seperti yang terlihat pada image 1

Jawaban :



1. Lalu lakukan pengecekan data dengan memilih menu 4

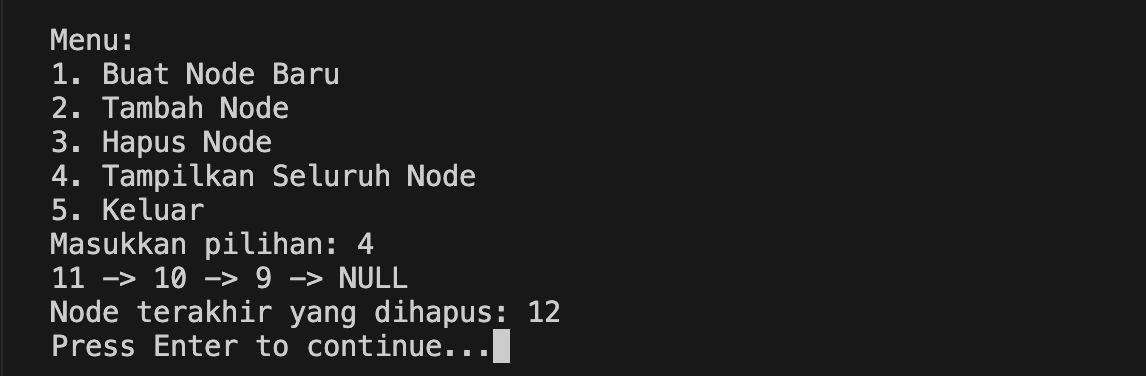


A computer chip with lines and dots

Description automatically generatedPenjelasan :

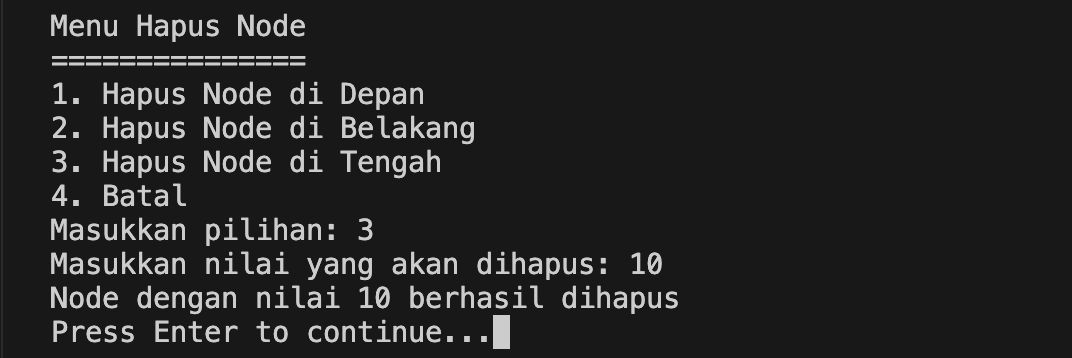
Pada tahapan ini, setelah kita melakukan penghapusan dan ingin menampilkan data, maka akan muncul list data dan menampilkan pemberitahuan bahwa data node yang terakhir dihapus adalah 12 dimana data 12 adalah data Node di depan seperti pada image 1

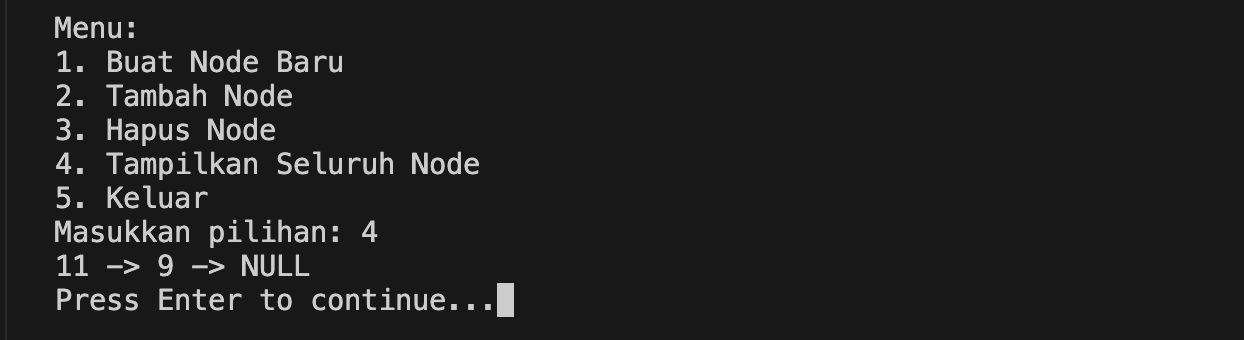
Jawaban :



1. Lakukan penghapusan data di tengah lalu tampilkan list data seperti pada image 1 pada tahapan 3 lalu tampilkan outputnya

Jawaban :

s



1. Lakukan penghapusan data di belakang lalu tampilkan list data seperti pada A computer chip with lines and dots

   Description automatically generatedimage 1 pada tahapan 3 lalu tampilkan outputnya

Jawaban :

