**LAPORAN PRAKTIKUM**

**STRUKTUR DATA**

**MODUL 5**

****

|  |
| --- |
| **Disusun Oleh:**  Nama: Juan Axl Ronaldio Zaka Putra  NIM: 220411100066  Kelas: IF 2D  **Dosen Pengampu:**  Nama: Hermawan, S.T., M.Kom.  NIP: 197908282005011002  **Asisten Praktikum:**  Nama: Moh. Fadil Abdillah  NIM: 210411100142 |

**PRODI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

**2023**

1. **Soal**

Tambahkan code diatas supaya bisa menambahkan node baru diawal linked list, minimal 7 node.

Contoh :

insert pertama (1), insert kedua (2), insert ketiga(3)

output : 3->2-> 1->

**Jawaban**

**Code Program:**

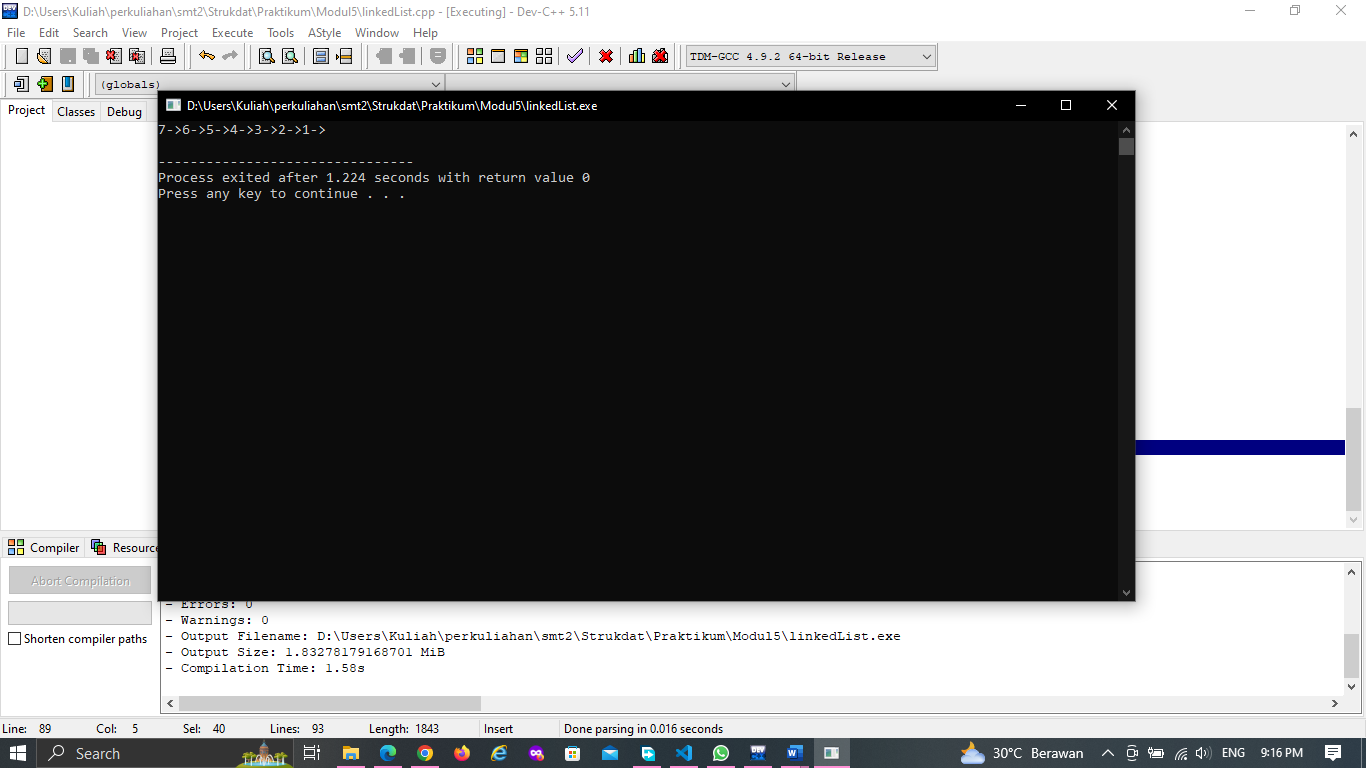
|  |
| --- |
| struct Node {  int data;  Node\* next;  };  void tambahNodeAwal(Node\*\* head, int dataBaru)  {  // Buat node baru  Node\* newNode = new Node;  newNode->data = dataBaru;  // Jika linked list masih kosong, node baru menjadi head  if (\*head == NULL) {  \*head = newNode;  newNode->next = NULL;  return;  }    // tambahkan node baru ke elemen pertama  newNode->next = \*head;  \*head = newNode;  }  void cetakList(Node\* head)  {  while (head != NULL) {  cout << head->data << "->";  head = head->next;  }  cout << endl;  }  int main()  {  Node\* head = NULL;  tambahNodeAwal(&head, 1);  tambahNodeAwal(&head, 2);  tambahNodeAwal(&head, 3);  tambahNodeAwal(&head, 4);  tambahNodeAwal(&head, 5);  tambahNodeAwal(&head, 6);  tambahNodeAwal(&head, 7);  cetakList(head);  return 0;  } |

**Penjelasan Code Program:**

Membuat fungsi void tambahNodeAwal(Node\*\* head, int dataBaru) untuk menambahkan node pada awal linked list dengan menggunakan parameter variabel pointer head dan integer dataBaru yaitu data yang akan dimasukkan. Lalu didalam fungsi tersebut membuat node baru pada variabel newNode dengan membuat objek Node baru. Melakukan pengkondisian jika variabel pointer head masih belum terisi atau NULL, maka head adalah newNode dan next dari newNode adalah NULL, lalu return agar fungsi langsung berhenti. Jika tidak memenuhi pengkondisian tersebut maka akan langsung menjalankan program yang diluar pengkondisian yaitu, next dari newNode adalah head dan head dipindahkan ke newNode.

Menjalankan program pada utama atau fungsi main(), membuat variabel head dengan tipe data struct Node dan diisikan NULL karena linked list masih kosong. Memanggil fungsi tambahNodeAwal(&head, int nilai) dan mengisikan parameternya dengan alamat dari head dan nilai yang akan dimasukkan untuk melakukan insert pada linked list, dilakukan sebanyak 7 kali dengan memasukkan nilai berurutan dari angka 1-7. Terakhir memanggil fungsi cetakList(head) yang telah dibuat sebelumnya dan mengisikan parameternya dengan variabel head untuk menampilkan setiap data pada linked list.

**Hasil Running Program:**



1. **Soal**

Buat code untuk menghapus node pada linked list di akhir.

**Jawaban**

**Code Program:**

|  |
| --- |
| struct Node {  int data;  Node\* next;  };  void tambahNodeAwal(Node\*\* head, int dataBaru)  {  // Buat node baru  Node\* newNode = new Node;  newNode->data = dataBaru;  // Jika linked list masih kosong, node baru menjadi head  if (\*head == NULL) {  \*head = newNode;  newNode->next = NULL;  return;  }  newNode->next = \*head;  \*head = newNode;  }  void hapusAkhir(Node\*\* head)  {  Node \*before, \*last = \*head;  while (last->next != NULL) {  before = last;  last = last->next;  }  before->next = NULL;  delete last;  }  void cetakList(Node\* head)  {  while (head != NULL) {  cout << head->data << "->";  head = head->next;  }  cout << endl;  }  int main()  {  Node\* head = NULL;  tambahNodeAwal(&head, 1);  tambahNodeAwal(&head, 2);  tambahNodeAwal(&head, 3);  tambahNodeAwal(&head, 4);  tambahNodeAwal(&head, 5);  tambahNodeAwal(&head, 6);  tambahNodeAwal(&head, 7);  cout << "Sebelum dilakukan hapusAkhir():" << endl;  cetakList(head);  hapusAkhir(&head);  cout << "\nSetelah dilakukan hapusAkhir():" << endl;  cetakList(head);  return 0;  } |

**Penjelasan Code Program:**

Membuat fungsi void hapusAkhir(Node\*\* head) untuk menghapus node terakhir pada linked list dengan parameter variabel pointer head. Lalu didalam fungsi tersebut mendeklarasikan variabel pointer bertipe data struct Node dengan nama before untuk nantinya menyimpan node sebelum node terakhir, dan last untuk nantinya menyimpan node terakhir. Variabel last diset ke node pertama terlebih dahulu atau head. Berikutnya melakukan traversing menggunakan while selama next dari last tidak bernilai NULL, pada while tersebut before diisikan last lalu last diubah ke last->next. Perulangan tersebut akan berhenti saat last berisi node terakhir atau node yang nextnya bernilai NULL, dan before berisi node sebelum node terakhir. Selanjutnya, next dari before akan diubah ke NULL agar menandakan kalau sudah tidak ada node selanjutnya lagi, dan melakukan delete pada variabel last.

Pada fungsi utama atau main(), melakukan penghapusan pada node terakhir yang telah di insertkan sebelumnya dengan memanggil fungsi hapusAkhir(&head) dan mengisikan parameternya dengan alamat dari variabel head untuk mengapus node terakhir. Terakhir memanggil fungsi cetakList(head) yang telah dibuat sebelumnya dan mengisikan parameternya dengan variabel head untuk menampilkan setiap data pada linked list.

**Hasil Running Program:**

