**LABORATORY PROGRAM – 9**

WAP to Implement doubly link list with primitive operations

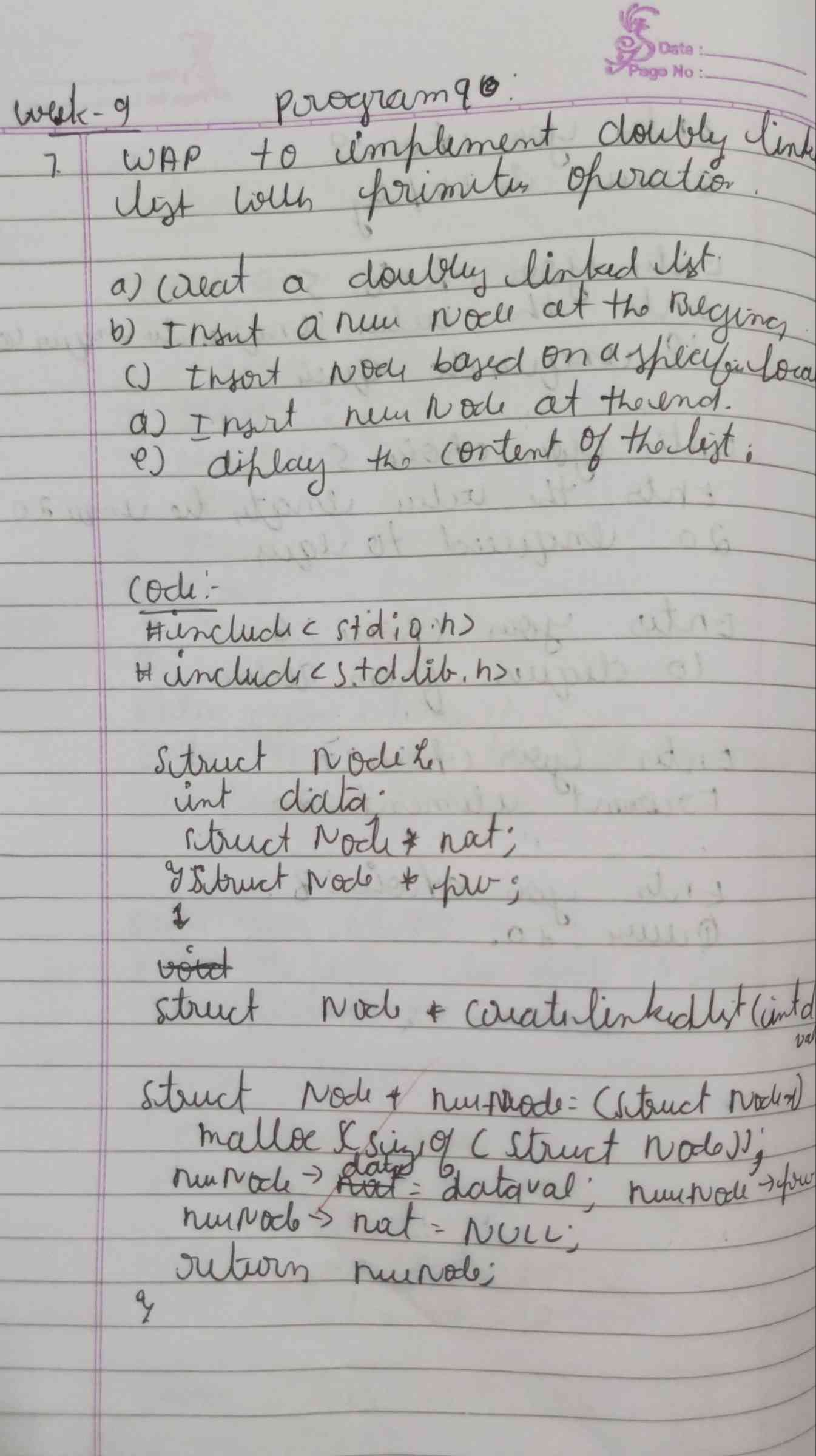
a) Create a doubly linked list.

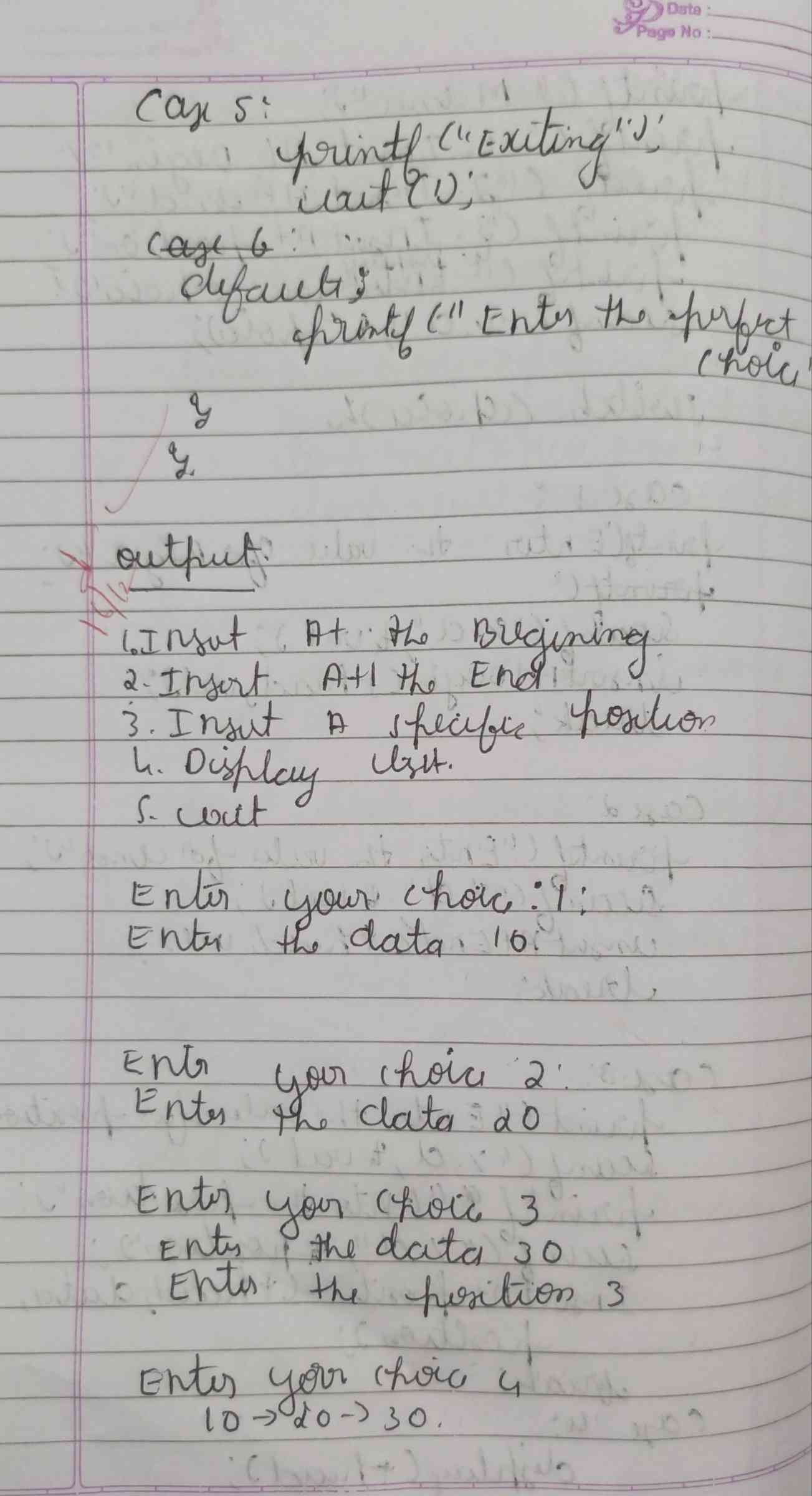
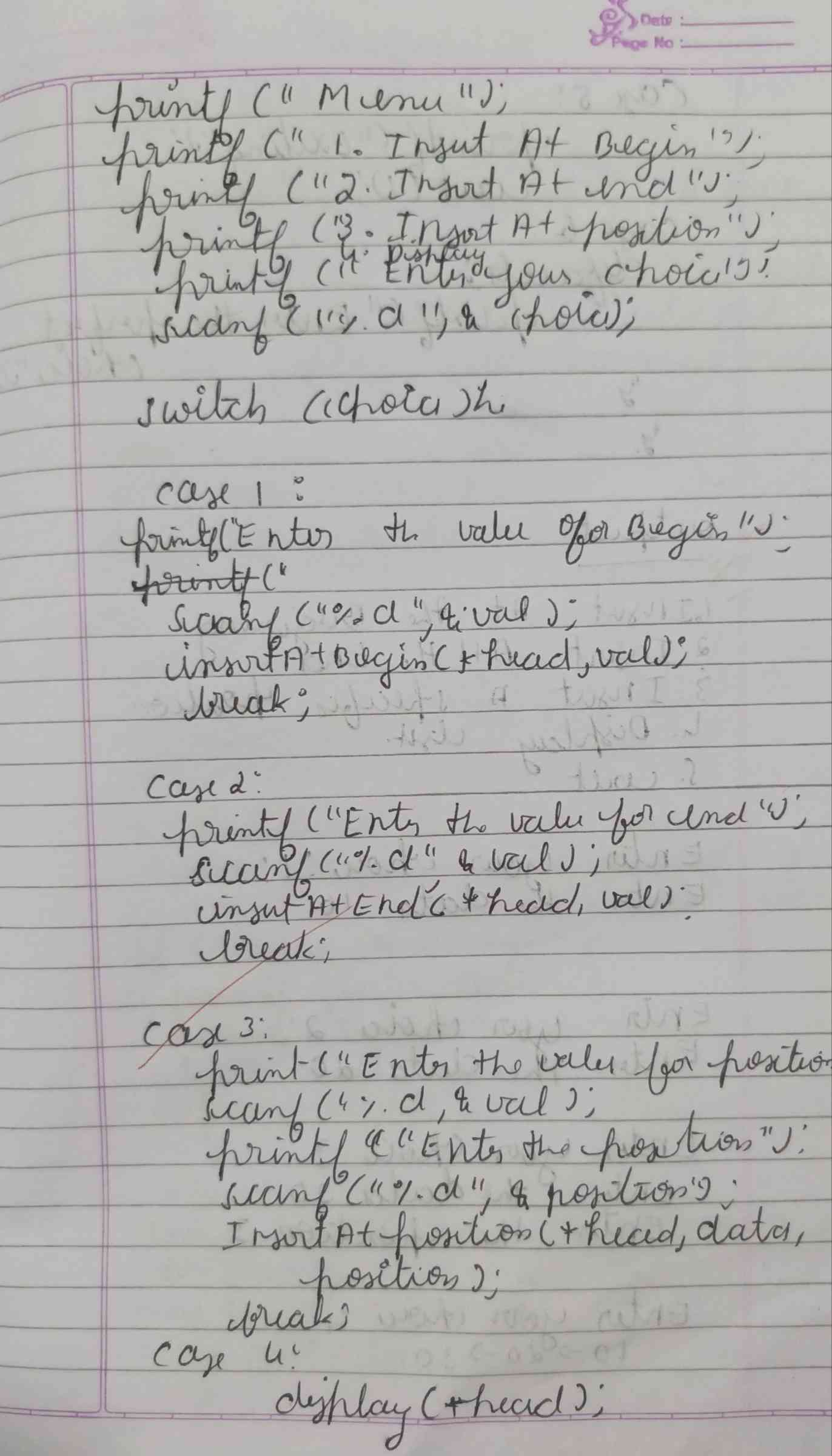
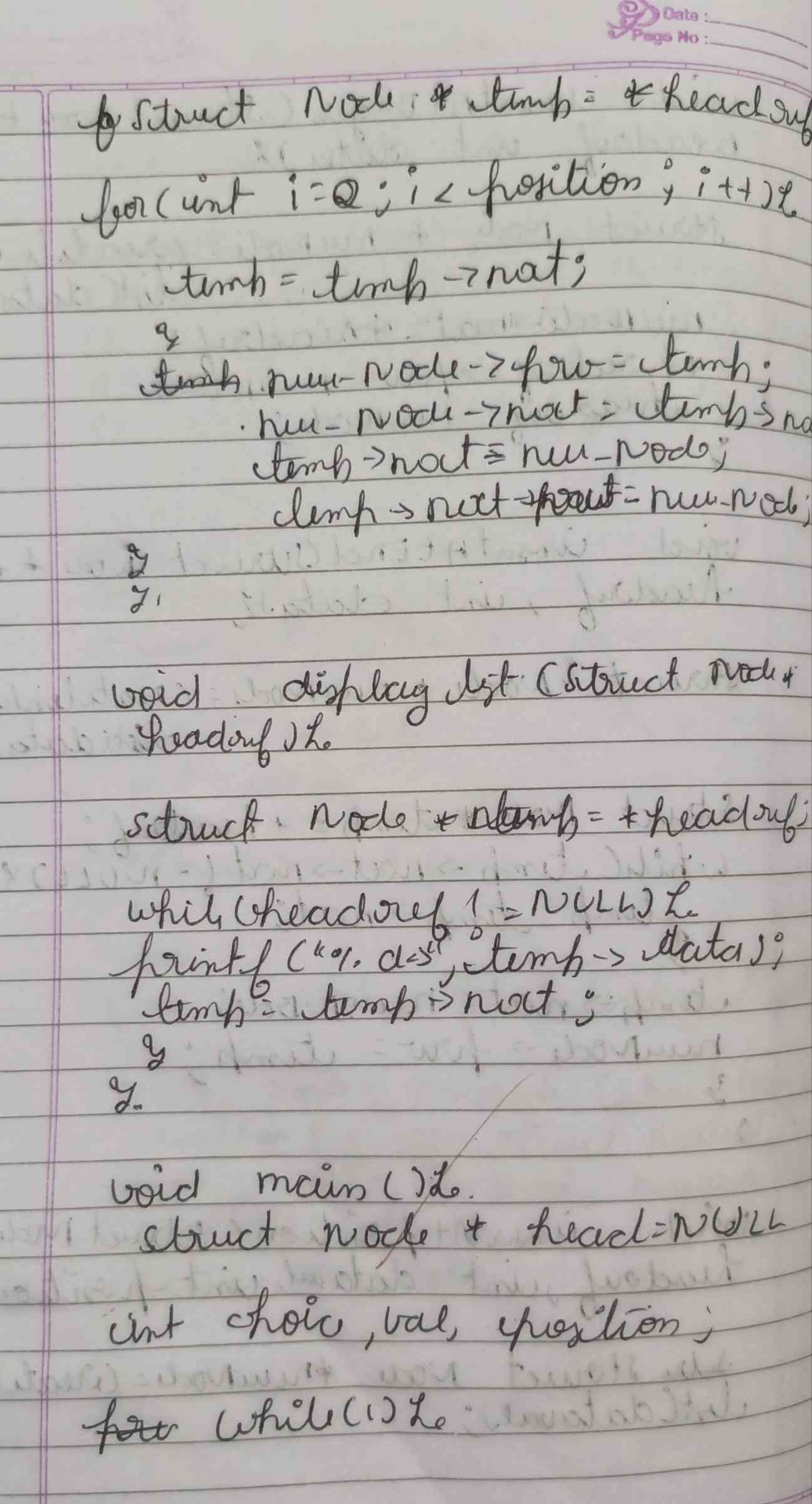
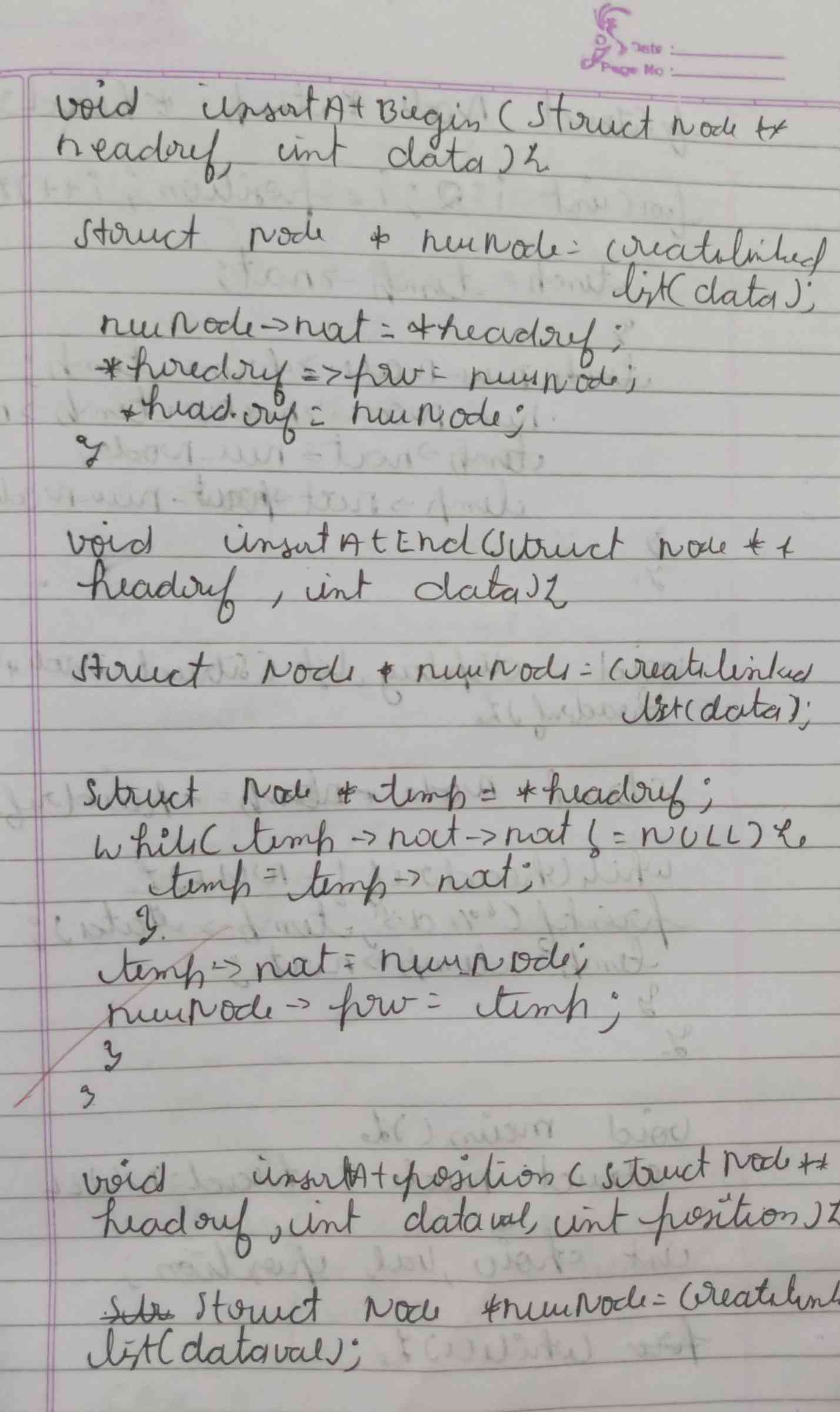
b) Insert a new node to the left of the node.

c) Delete the node based on a specific value

d) Display the contents of the list

**OBSERVATION :**

****



**CODE :**

**#include <stdio.h>**

**#include <stdlib.h>**

**struct Node {**

**int data;**

**struct Node\* next;**

**struct Node\* prev;**

**};**

**struct Node\* createnode(int data) {**

**struct Node\* newNode = (struct Node\*)malloc(sizeof(struct Node));**

**newNode->data = data;**

**newNode->next = NULL;**

**newNode->prev = NULL;**

**return newNode;**

**}**

**void insertatbegin(struct Node\*\* head, int data) {**

**struct Node\* newNode = createnode(data);**

**if (\*head == NULL) {**

**\*head = newNode;**

**} else {**

**newNode->next = \*head;**

**(\*head)->prev = newNode;**

**\*head = newNode;**

**}}**

**void insertatend(struct Node\*\* head, int data) {**

**struct Node\* newNode = createnode(data);**

**if (\*head == NULL) {**

**\*head = newNode;**

**} else {**

**struct Node\* temp = \*head;**

**while (temp->next != NULL) {**

**temp = temp->next;**

**}**

**temp->next = newNode;**

**newNode->prev = temp;**

**}}**

**void insertatposition(struct Node\*\* head, int data, int position) {**

**struct Node\* newNode = createnode(data);**

**if (position == 1) {**

**insertatbegin(head, data);**

**return;**

**}**

**struct Node\* temp = \*head;**

**for(int i=2;i<position;i++){**

**temp = temp->next;**

**}**

**newNode->next = temp->next;**

**newNode->prev = temp;**

**if (temp->next != NULL) {**

**temp->next->prev = newNode;**

**}**

**temp->next = newNode;**

**}**

**void display(struct Node\* head) {**

**struct Node\* temp = head;**

**if (temp == NULL) {**

**printf("List is empty.\n");**

**return;   }**

**while (temp != NULL) {**

**printf("%d -> ", temp->data);**

**temp = temp->next;**

**}**

**printf("\n"); }**

**int main() {**

**struct Node\* head = NULL;**

**int choice, data, position;**

**while (1) {**

**printf("1. Insert at the beginning\n");**

**printf("2. Insert at the end\n");**

**printf("3. Insert at a specific position\n");**

**printf("4. Display list\n");**

**printf("5. Exit\n");**

**printf("Enter your choice: ");**

**scanf("%d", &choice);**

**switch (choice) {**

**case 1:**

**printf("Enter the data  ");**

**scanf("%d", &data);**

**insertatbegin(&head, data);**

**break;**

**case 2:**

**printf("Enter the data  ");**

**scanf("%d", &data);**

**insertatend(&head, data);**

**break;**

**case 3:**

**printf("Enter the data: ");**

**scanf("%d", &data);**

**printf("Enter the position : ");**

**scanf("%d", &position);**

**insertatposition(&head, data, position);**

**break;**

**case 4:**

**display(head);**

**break;**

**case 5:**

**printf("Exit\n");**

**exit(0);**

**default:**

**printf("Invalid choice\n");**

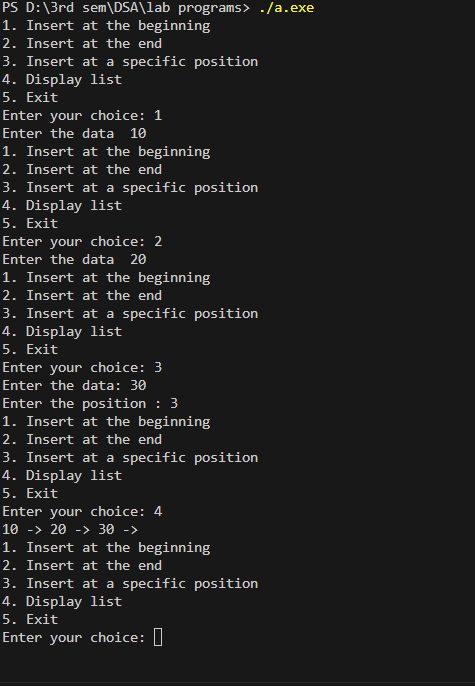
**}**

**}**

**return 0;**

**}**

**OUTPUT :**

****