

Nama : Khusnia Fitri

NIM : 1203230030

Code:

```
#include <stdio.h>
#include <stdlib.h>

typedef struct Node {
    int data;
    struct Node* prev;
    struct Node* next;
} Node;

Node* createNode(int data) {
    Node* newNode = (Node*)malloc(sizeof(Node));
    newNode->data = data;
    newNode->prev = newNode->next = NULL;
    return newNode;
}

int contains(Node* head, int data) {
    if (head == NULL) return 0;
    Node* temp = head;
    do {
        if (temp->data == data) return 1;
        temp = temp->next;
    } while (temp != head);
    return 0;
}

void insertEnd(Node** head, int data) {
    if (contains(*head, data)) return;
    Node* newNode = createNode(data);
    if (*head == NULL) {
        *head = newNode;
        newNode->next = newNode->prev = newNode;
    } else {
        Node* last = (*head)->prev;
        newNode->next = *head;
        (*head)->prev = newNode;
        newNode->prev = last;
        last->next = newNode;
    }
}
```

```

void printList(Node* head) {
    if (head == NULL) return;
    Node* temp = head;
    do {
        printf("Address: %p, Data: %d\n", (void*)temp, temp->data);
        temp = temp->next;
    } while (temp != head);
}

```

```

void sortList(Node** head) {
    if (*head == NULL) return;
    Node *temp1, *temp2;
    int swapped;
    do {
        swapped = 0;
        temp1 = *head;
        while (temp1->next != *head) {
            temp2 = temp1->next;
            if (temp1->data > temp2->data) {
                temp1->prev->next = temp2;
                temp2->prev = temp1->prev;
                temp1->next = temp2->next;
                temp2->next->prev = temp1;
                temp1->prev = temp2;
                temp2->next = temp1;
                if (*head == temp1)
                    *head = temp2;
                swapped = 1;
            } else {
                temp1 = temp1->next;
            }
        }
    } while (swapped);
}

```

```

int main() {
    Node* head = NULL;
    int N, Ai;
    printf("Masukkan jumlah data (N): ");
    scanf("%d", &N);
    printf("");
    for (int i = 0; i < N; i++) {
        scanf("%d", &Ai);
        insertEnd(&head, Ai);
    }
}

```

```
}  
printf("List sebelum diurutkan:\n");  
printList(head);  
sortList(&head);  
printf("List setelah diurutkan:\n");  
printList(head);  
return 0;  
}
```

Penjelasan :

1. Struktur Node: memiliki tiga anggota: data yang menyimpan nilai data, prev yang menunjukkan ke node sebelumnya, dan next yang menunjukkan ke node berikutnya.
2. Fungsi createNode: digunakan untuk membuat node baru dengan alokasi memori dinamis dan mengatur nilai-nilai awalnya.
3. Fungsi contains: untuk memeriksa apakah data sudah ada dalam list. Jika data sudah ada, fungsi akan mengembalikan 1; jika tidak, fungsi akan mengembalikan 0.
4. Fungsi insertEnd: Fungsi ini menambahkan node baru ke ujung list, tetapi hanya jika data tersebut belum ada dalam list.
5. Fungsi printList: Fungsi ini mencetak seluruh list, dimulai dari head hingga kembali ke head.
6. Fungsi sortList: Fungsi ini mengurutkan list secara ascending dengan menggunakan algoritma Bubble Sort, algoritma ini hanya mengubah posisi dari node-node tersebut tanpa memanipulasi data pada setiap node.