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

struct node { int value;

struct node \*next;

};

typedef struct node \*NODE;

NODE get\_node() {

NODE ptr = (NODE)malloc(sizeof(struct node)); if (ptr == NULL) {

printf("Memory not allocated\n");

}

return ptr;

}

NODE delete\_first(NODE first){ NODE temp=first;

if (first == NULL) { printf("Empty\n"); return NULL;

}

first=first->next; free(temp); return first;

}

NODE delete\_end(NODE first){ if (first == NULL) {

printf("Empty\n"); return NULL;

}

NODE prev,last; prev=NULL; last=first;

while(last->next!=NULL){ prev=last;

last=last->next;

}

prev->next=NULL;

free(last); return first;

}

NODE delete\_value(NODE first,int value){ if (first == NULL) {

printf("Empty\n"); return NULL;

}

NODE prev,current; prev=NULL; current=first;

while(value!=current->value || current->next!=NULL){ prev=current;

current=current->next;

}

if(current==NULL){ printf("Value notfound"); return first;

}

prev->next=current->next; free(current);

return first;

}

NODE insert\_beginning(NODE first, int item) { NODE new\_node = get\_node();

new\_node->value = item; new\_node->next = first; return new\_node;

}

void display(NODE first) { NODE temp = first;

if (first == NULL) { printf("Empty\n"); return;

}

while (temp != NULL) { printf("%d ", temp->value); temp = temp->next;

}

printf("\n");

}

int main() {

int item, choice; NODE first = NULL;

first = insert\_beginning(first, 6); first = insert\_beginning(first, 5); first = insert\_beginning(first, 4); first = insert\_beginning(first, 3); first = insert\_beginning(first, 2); first = insert\_beginning(first, 1);

printf("List before deleting:\n"); display(first);

while (1) {

printf("Choose an operation to delete:\n"); printf("1. Delete at beginning\n"); printf("2. Delete at end\n");

printf("3. Delete specific value\n"); printf("4. Display list\n"); printf("5. Exit\n");

scanf("%d", &choice);

switch (choice) { case 1:

printf("Deleting at first.\n"); first = delete\_first(first); display(first);

break; case 2:

printf("Deleting at the end.\n"); first = delete\_end(first); display(first);

break; case 3:

printf("Enter value to delete: "); scanf("%d", &item);

first = delete\_value(first, item); display(first);

break; case 4:

printf("List: "); display(first);

break; case 5:

exit(0); default:

printf("Invalid choice. Please try again.\n");

}

}

return 0;

}