#include <stdio.h>

#include <stdlib.h>

typedef struct node

{

int data;

struct node \*next;

struct node \*prev;

}NODE;

NODE\* add(NODE\*, int);

void disp(NODE\*);

//int search(NODE\*, int);

int main()

{

//Create an empty warehouse

NODE \*head=NULL;

int opt, item;

while(1)

{

printf("\n1: Add item 2: Display ");

printf("3: exit");

printf("\nEnter your option: ");

scanf("%d", &opt);

switch(opt)

{

case 1: printf("\nEnter item to add to list: ");

scanf("%d", &item);

head=add(head, item);

break;

case 2: disp(head);

break;

case 3: exit(0);

}

}

return 0;

}

NODE\* add(NODE\* head, int item)

{

NODE \*start, \*curr;

NODE \*newnode=(NODE\*)malloc(sizeof(NODE));

if(newnode==NULL)

{

printf("\nMalloc failure");

exit(1);

}

newnode->data=item;

newnode->next=NULL;

newnode->prev=NULL;

// Case i - List is empty

if(head==NULL)

head=newnode;

else // Case ii - adding the smallest item

if(item < head->data)

{

newnode->next = head;

head = newnode;

}

else // Case iii

{

start=head;

curr=head->next;

while(curr && item > curr->data)

{

start=start->next;

curr=curr->next;

}//end of while

newnode->next=curr;

newnode->prev=start;

start->next=newnode;

}//end of else

return head;

}

void disp(NODE \*head)

{

if(head==NULL)

{

printf("\nlist is empty");

return;

}

printf("\nThe list items are: ");

while(head)

{

printf("%d ", head->data);

head=head->next;

}

}