10 5 9

Current=10

1

Previous=10

Current->next=5

Min=5

2

Prevminptr=previous (10)

Minptr=5

Previous=10

Current->next=9

3

Pervious=9

Current->next=null(end of list)

Previousminptr->next=minptr->next (10 -> 9)

Minptr->next=head (5 -> 10)

Head=minptr (5 is now the head of the list)

5 10 9

#include<stdio.h>

#include<stdlib.h>

typedef struct Node

{

int num;

Node \*next;

}Node;

Node \*makenode(int num)

{

Node \*new\_node = (Node\*)malloc(sizeof(Node));

if (new\_node==NULL)

{

printf("error creating a new node\n");

exit(0);

}

new\_node->num = num;

new\_node->next = NULL;

return new\_node;

}

void addnode(Node \*head, Node \*new\_node)

{

Node \*current = head;

while (current->next!=NULL)

{

current = current->next;

}

current->next = new\_node;

}

void printlist(Node \*head)

{

Node \*current = head;

while (current->next != NULL)

{

printf("%d ", current->num);

current = current->next;

if (current->next == NULL)

{

printf("%d ", current->num);

}

}

}

Node \*func(Node \*head)

{

int min = head->num;

Node \*current = head, \*minPtr = NULL, \*previous = NULL, \*prevMinPtr = NULL;

while (current)//going over the list

{

if (current->num < min)//check for the smallest number

{

min = current->num;//smallest number only

prevMinPtr = previous;//save the previous element(the one before current)

minPtr = current;//define the current elemnt as the smallest element

}

previous = current;//save the previous element(the one before current)

current = current->next;//go on the next element

}

if (head == minPtr)//if the first element is the smallest one, define it as the head of the list and return it

return head;

prevMinPtr->next = minPtr->next;//define that the two previous elements will point to each other without changing their order

minPtr->next = head;//define that the smallest number will point to the head

head = minPtr;//define the smallest element as the first element

printf("\n===========\n");

printlist(head);//print the new list(a function)

return head;

}

void main()

{

Node \*head;

Node \*new\_node;

int num,i;

printf("enter num:\n");

scanf("%d", &num);

head = makenode(num);

for ( i = 1; i < 3; i++)

{

printf("enter num:\n");

scanf("%d", &num);

new\_node = makenode(num);

addnode(head, new\_node);

}

printlist(head);

func(head);

}