

```

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

#define TAM 20

typedef struct _lista{
    int valor;
    struct _lista *prox;
}LISTA;

void imprimirLista(LISTA *head);

LISTA *novoNO(int valor)
{
    LISTA *novo = (LISTA *)malloc(sizeof(LISTA));
    novo->prox = NULL;
    novo->valor = valor;
    return novo;
}

void inserir(LISTA *head,int valor)
{
    LISTA *novo = novoNO(valor);
    if (head->prox==NULL)
    {
        head->prox=novo;
    }
    else
    {
        LISTA *aux = head->prox;
        LISTA *ant = head;
        while (aux!=NULL && valor > aux->valor)
        {
            ant = aux;
            aux=aux->prox;
        }
        if (ant==head)
        {
            head->prox = novo;
            novo->prox = aux;
        }
        else
    }

```

```

    {
        ant->prox = novo;
        novo->prox = aux;
    }
    imprimirLista(head);
    /*LISTA *aux = head->prox;
    while (aux->prox!=NULL && valor < aux->prox->valor)
    {
        aux=aux->prox;
    }
    if (aux==head->prox)
    {
        head->prox = novo;
        novo->prox = aux;
    }
    else
    {
        novo->prox = aux->prox;
        aux->prox = novo;
    }*/
}
}

```

```

void imprimirLista(LISTA *head)
{
    LISTA *aux;
    printf("Lista: ");
    for (aux=head->prox;aux!=NULL;aux=aux->prox)
    {
        printf("%i->",aux->valor);
    }
    printf("\n");
}

```

```

void liberaLista(LISTA *head)
{
    if (head!=NULL)
    {
        liberaLista(head->prox);
        free(head);
    }
}

```

```
int main()
{
    LISTA *head = (LISTA *)malloc(sizeof(LISTA));
    head->prox = NULL;
    srand(time(NULL));

    int i,valor;
    for (i=0;i<TAM;i++)
    {
        valor = (rand()%100)-50;
        inserir(head,valor);
    }
    imprimirLista(head);
    liberaLista(head);

    return 0;
}
```