1/3/2021

Hande Özen

Sakarya üniversitesi

Hande Özen

Bilişim Sistemleri Mühendisliği

B181200048

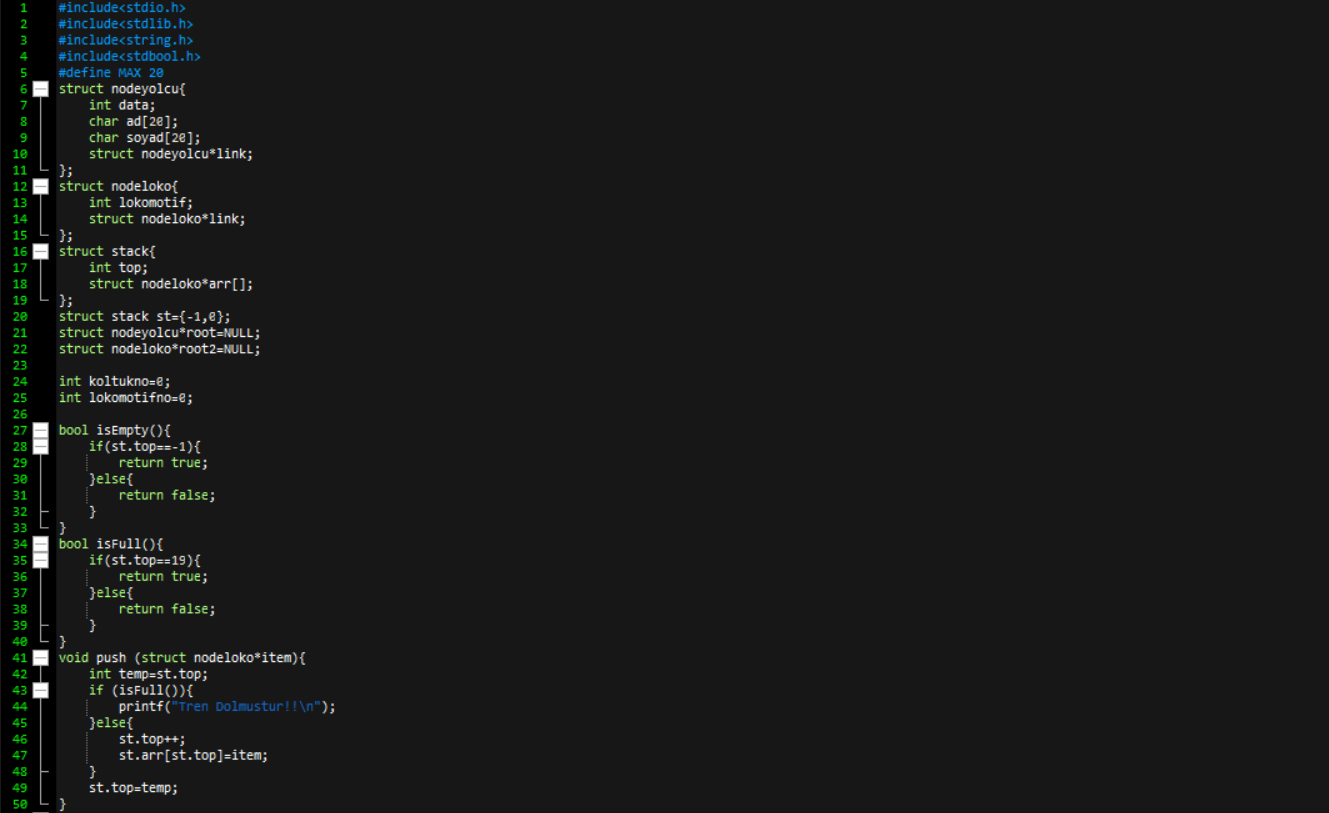
***Ödevin Konusu:*** Her bir lokomotifinde 10 tane koltuk olduğu varsayılan bir trene maksimum 20 adet lokomotif eklenebilmektedir. (Lokomotif ekleme işlemi lokomotif dolduktan sonra yapılabilmeltedir) Yolcuların sadece ilk lokomotiften trene binebildikleri ve her bir lokomotifte boş yer bırakmadan sıra ile koltuklara oturabildikleri bir yapıyı yığın ve liste kullanarak tasarlayıp c, c++ dilinde kodlayınız.

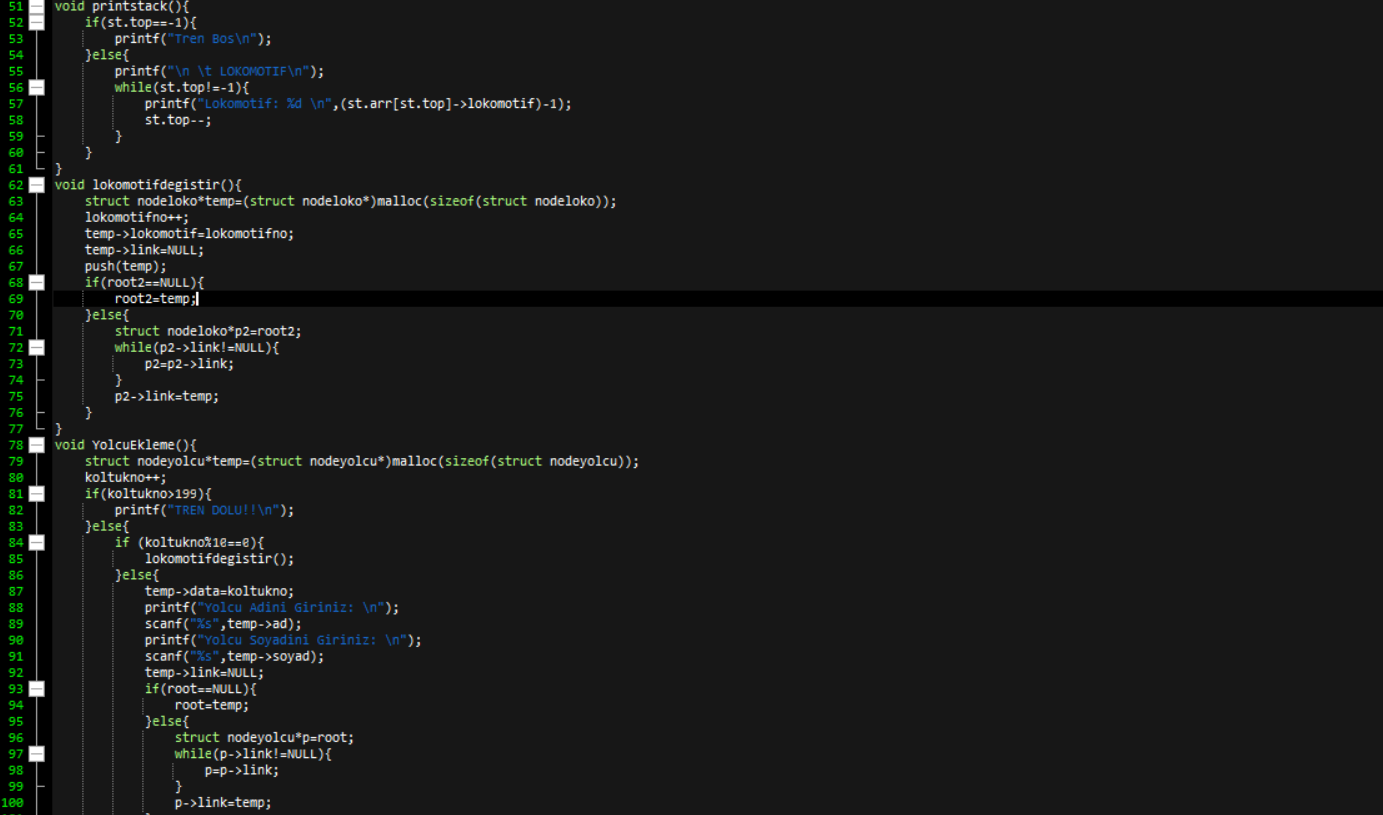
**2020-2021 Güz Dönemi VERİ YAPILARI 2.VE 3. ÖDEVİ**

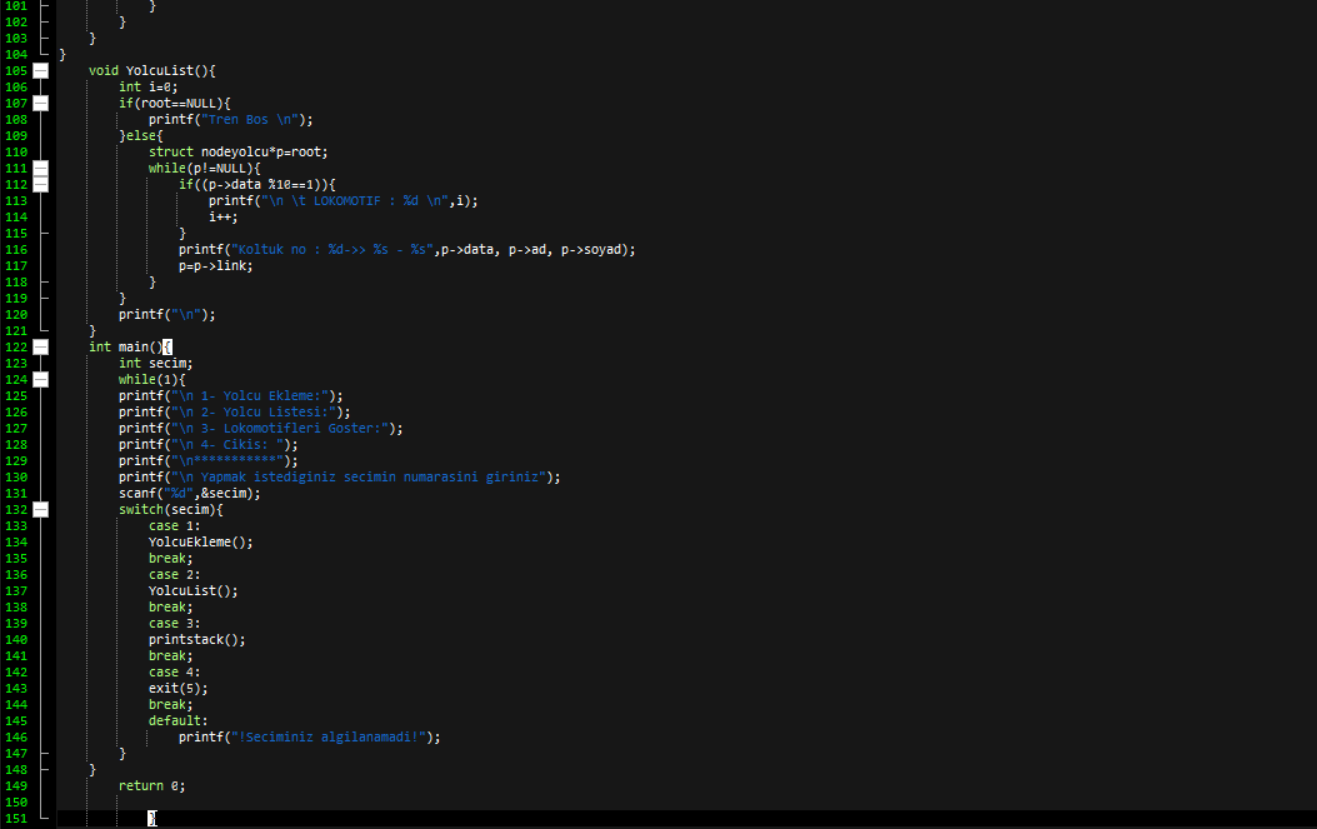
Can Yüzkollar

**VİDEO LİNK: https://www.youtube.com/watch?v=UErnEqBIEcc**

**KOD:**







#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<stdbool.h>

#define MAX 20

struct nodeyolcu{

int data;

char ad[20];

char soyad[20];

struct nodeyolcu\*link;

};

struct nodeloko{

int lokomotif;

struct nodeloko\*link;

};

struct stack{

int top;

struct nodeloko\*arr[];

};

struct stack st={-1,0};

struct nodeyolcu\*root=NULL;

struct nodeloko\*root2=NULL;

int koltukno=0;

int lokomotifno=0;

bool isEmpty(){

if(st.top==-1){

return true;

}else{

return false;

}

}

bool isFull(){

if(st.top==19){

return true;

}else{

return false;

}

}

void push (struct nodeloko\*item){

int temp=st.top;

if (isFull()){

printf("Tren Dolmustur!!\n");

}else{

st.top++;

st.arr[st.top]=item;

}

st.top=temp;

}

void printstack(){

if(st.top==-1){

printf("Tren Bos\n");

}else{

printf("\n \t LOKOMOTIF\n");

while(st.top!=-1){

printf("Lokomotif: %d \n",(st.arr[st.top]->lokomotif)-1);

st.top--;

}

}

}

void lokomotifdegistir(){

struct nodeloko\*temp=(struct nodeloko\*)malloc(sizeof(struct nodeloko));

lokomotifno++;

temp->lokomotif=lokomotifno;

temp->link=NULL;

push(temp);

if(root2==NULL){

root2=temp;

}else{

struct nodeloko\*p2=root2;

while(p2->link!=NULL){

p2=p2->link;

}

p2->link=temp;

}

}

void YolcuEkleme(){

struct nodeyolcu\*temp=(struct nodeyolcu\*)malloc(sizeof(struct nodeyolcu));

koltukno++;

if(koltukno>199){

printf("TREN DOLU!!\n");

}else{

if (koltukno%10==0){

lokomotifdegistir();

}else{

temp->data=koltukno;

printf("Yolcu Adini Giriniz: \n");

scanf("%s",temp->ad);

printf("Yolcu Soyadini Giriniz: \n");

scanf("%s",temp->soyad);

temp->link=NULL;

if(root==NULL){

root=temp;

}else{

struct nodeyolcu\*p=root;

while(p->link!=NULL){

p=p->link;

}

p->link=temp;

}

}

}

}

void YolcuList(){

int i=0;

if(root==NULL){

printf("Tren Bos \n");

}else{

struct nodeyolcu\*p=root;

while(p!=NULL){

if((p->data %10==1)){

printf("\n \t LOKOMOTIF : %d \n",i);

i++;

}

printf("Koltuk no: %d->> %s - %s",p->data, p->ad, p->soyad);

p=p->link;

}

}

printf("\n");

}

int main(){

int secim;

while(1){

printf("\n 1- Yolcu Ekleme:");

printf("\n 2- Yolcu Listesi:");

printf("\n 3- Lokomotifleri Goster:");

printf("\n 4- Cikis: ");

printf("\n\*\*\*\*\*\*\*\*\*\*\*");

printf("\n Yapmak istediginiz secimin numarasini giriniz");

scanf("%d",&secim);

switch(secim){

case 1:

YolcuEkleme();

break;

case 2:

YolcuList();

break;

case 3:

printstack();

break;

case 4:

exit(5);

break;

default:

printf("!Seciminiz algilanamadi!");

}

}

return 0;

}

**EKRAN GÖRÜNTÜSÜ:**

****

Yolcu ekleme işlemi yapıldıktan sonra yolcu listesinden eklenen yolcuların koltuk numaraları sırasıyla gösterilir.