```c

char v[16][20]={ // declaracao de array global
"",//0
"Aracajú",//1
"Belém",//2
"Belo Horizonte",//3
"Brasília",//4
"Florianópolis",//5
"Fortaleza",//6
"Goiânia",//7
"João Pessoa",//8
"Natal",//9
"Porto Alegre",//10
"Recife",//11
"Rio de Janeiro",//12
"Salvador",//13
"São Luiz",//14
"São Paulo"//15
};

typedef struct Tdados{ //registro
    int idia;//dia inicial do sistema
    int imes;//mes inicial do sistema
    int iano;//ano inicial do sistema
    int dia;
    int mes;
    int ano;
    char nome[50];
    char data[11];
    int origem;
    int destino;
    int contvoo;//contador de voo
    int contespera;//contador de fila de espera
    char lista[16][50]; //lista de passageiros para o voo e fila de espera
}Tdados;// registro de dados

Tdados IniciaTdados(){//metodo de inicializador
    int i;
    Tdados aux;
    aux.idia=0;
    aux.imes=0;
    aux.iano=0;
    aux.dia=0;
    aux.mes=0;
    aux.ano=0;
    aux.contvoo=0;
    aux.contespera=0;
    strcpy(aux.nome,"");//atribuindo vazio a string
    strcpy(aux.data,"");//atribuindo vazio a string
    aux.origem=0;
    aux.destino=0;

    for(i=0;i<16;i++){//limpa de 0 a 15
    strcpy(aux.lista[i],"");//atribuindo vazio a string de cada posicao do vetor
    }
    return aux;
}

void espacos(){ // pular linhas
int i;
    for(i=0;i<5;i++){
        printf("\n");
    }
}

void listaaeroportos(int exclui){///imprime a lista de aeroportos
 int i;//variavel de contro do for
 printf("-------------------------\n");
 for(i=1;i<=15;i++){//exclui recebe o valor de 1 a 15 de origem ou 0 se for usado para origem
    if(i!=exclui){//imprime o conteudo do vetor de 1 a 15 que for diferente do valor de exclui
    printf("%d %s \n",i,v[i]);
    }
 } //lista de aeroportos
 printf("-------------------------\n");
}

char *valornome(){//captura o valor de voo
    espacos();// inserir alguns enter na tela
    char nome[50];
    printf("----------------------------------------\n");
    printf(" Nome do passageiro .: \n");
    printf("----------------------------------------\n");
    fflush(stdin);//limpar buffer de entrada antes da leitura
    gets(nome);//capturar variavel
    strupr(nome);//deixar string maiuscula
    fflush(stdin);//limpar buffer depois da leitura
    espacos();//inserir alguns enter na tela

    return nome;//returna valor da funcao
}

char *validanome(){//validacao de nome
char nome[50];
int flag;
flag=0;
do{//repita enquanto o nome estiver VAZIO
    system("cls");
if(flag==1){//se tiver vazio aparece essa mensagem
    printf("\n\n\n----------------------------------------\n");
    printf(" !!! O CAMPO NOME NÃO PODE SER VAZIO !!!\n");
    printf("----------------------------------------\n");
}
strcpy(nome,valornome());//atribui o valor retornado da funcao para nome

if(strcmp(nome,"")==0){//se tiver vazio flag recebe 1
flag=1;
}else{//se nao tiver vazio flag recebe zero
    flag=0;
    }
}while(flag==1);//compara se o nome esta vazio enquanto for 1 repete

return nome;//retorno da funcao
}

int valororigem(){//captura valor de origem
    int origem;
    listaaeroportos(0);//exibe lista de aeroportos SEM exclusao de nomes
    printf("-----------------------------------------\n");
    printf(" Informe o numero correspondente a origem .: \n");
    printf("-----------------------------------------\n");
    scanf("%d",&origem);//captura o valor para origem
    fflush(stdin);//limpar buffer depois da leitura
    espacos();//inserir enter na tela

    return origem;//retorno da funcao
}

int validaorigem(){//validacao de origem
int valor,flag;
flag=0;
    do{//repita ate que o valor seja entre 1 e 15

        if(flag==1){//se repitiu aparece a mensagem ...
            system("cls");
            printf("\n\n\n----------------------------------------------------------------\n");
            printf(" !!! O CAMPO ORIGEM DEVE TER APENAS UM DOS NUMEROS LISTADOS !!!\n");
            printf("----------------------------------------------------------------\n");
        }
        valor=valororigem();//atribui valor retornado da funcao para a variavel valor

        if(valor<1 || valor>15){//se o  valor eh menor que 1 ou maior que 15 repete
            flag=1;
        }else{//senao sai do loop
            flag=0;
        }
    }while(flag==1);
return valor;//retorno da funcao
}

int valordestino(int origem){//captura valor de destino
    int destino,flag;
    flag=0;
    do{//REPITA enquanto DESTINO for igual a ORIGEM

        if(flag==1){//se repitiu aparece a mensagem ...
            system("cls");//limpa tela
            printf("\n\n\n----------------------------------------------------------------\n");
            printf(" !!! O CAMPO DESTINO NÃO PODE SER IGUAL A ORIGEM !!!\n");
            printf("----------------------------------------------------------------\n");
        }
        listaaeroportos(origem);//mostra lista de aeroportos excluindo a origem
        printf("-----------------------------------------\n");
        printf(" Informe o numero correspondente ao destino desejado .: \n");
        printf("-----------------------------------------\n");
        scanf("%d",&destino);//captura valor de destino
        fflush(stdin);//limpar buffer depois da leitura
        espacos();//inserir enter na tela

        if(origem==destino){//se o valor de origem é igual a destino flag recebe
            flag=1;
        }else{//senao sai do loop
            flag=0;
        }

    }while(flag==1);//repete enquanto origem for igual a destino
return destino;//retorno da funcao
}

int validadestino(int dest){//validacao de destino
int valor,flag;
    flag=0;
    do{//REPITA ate que destino seja entre 1 e 15
        if(flag==1){//se repitiu aparece a mensagem ...
            printf("\n\n\n----------------------------------------------------------------\n");
            printf(" !!! O CAMPO DESTINO DEVE TER APENAS UM DOS NUMEROS LISTADOS !!!\n");
            printf("----------------------------------------------------------------\n");

        }
    valor=valordestino(dest);//recebe retorno da funcao de destino
    if(valor<1 || valor>15){//se o  valor eh menor que 1 ou maior que 15 repete
        flag=1;
    }else{//senao sai do loop
        flag=0;
    }
    }while(flag==1);//sai do loop somente quando for DIFERENTE de 1
return valor;//retorno da funcao
}

char *valordata(){//captura data em formato string
    char data[11];
    printf("\nInforme a data desejada: ");
    scanf("%s",data);
    fflush(stdin); //limpar buffer de entrada
    espacos(); //insere enter na tela
return data;
}

void stringtodata(Tdados *aux){//data em string
  int i;
  char data[11],dia[3],mes[3],ano[5];
  printf ("Enter a number: ");
  fgets ( data, 11, stdin );//captura string (variavel, tamanho da string, comando )
    dia[0]=data[0];//copia o caracter da posicao especifica de uma string para outra
    dia[1]=data[1];//copia o caracter da posicao especifica de uma string para outra
    dia[2]='\0';//adiciona nulo para a posicao da string
    mes[0]=data[2];//copia o caracter da posicao especifica de uma string para outra
    mes[1]=data[3];//copia o caracter da posicao especifica de uma string para outra
    mes[2]='\0';//adiciona nulo para a posicao da string
    ano[0]=data[4];//copia o caracter da posicao especifica de uma string para outra
    ano[1]=data[5];//copia o caracter da posicao especifica de uma string para outra
    ano[2]=data[6];//copia o caracter da posicao especifica de uma string para outra
    ano[3]=data[7];//copia o caracter da posicao especifica de uma string para outra
    ano[4]='\0';//adiciona nulo para a posicao da string
    strcpy(data,dia);//copia dia para data
```

```
      strcat(data,"/");//concatena (adiciona, soma) a string o segundo valor imposto dentro dos parenteses
      strcat(data,mes);//concatena (adiciona, soma) a string o segundo valor imposto dentro dos parenteses
      strcat(data,"/");//concatena (adiciona, soma) a string o segundo valor imposto dentro dos parenteses
      strcat(data,ano);//concatena (adiciona, soma) a string o segundo valor imposto dentro dos parenteses
      printf("%s",dia);//concatena (adiciona, soma) a string o segundo valor imposto dentro dos parenteses
      aux->dia = atoi (dia);//converte inteiro para string
      aux->mes = atoi (mes);//converte inteiro para string
      aux->ano = atoi (ano);//converte inteiro para string

    printf (" %d %d %d %s\n",aux->dia,aux->mes,aux->ano,data);
}

void mostradigitadodata(Tdados *aux,int flag){//mostra data digitada até o momento
     char v1[2],v2[2],v3[4];//variaveis de auxilio
system("cls");//limpa tela
espacos();//coloca enter na tela

if(aux->dia<10){ strcpy(v1,"0"); }// se o dia é menor que 10 variavel de auxilio v1 recebe "0"
else{ strcpy(v1,"");}// senao a variavel de auxilio v1 fica vazia

if(aux->mes<10){strcpy(v2,"0");}// se o mes é menor que 10 variavel de auxilio v2 recebe "0"
else{ strcpy(v2,"");}// senao a variavel de auxilio v2 fica vazia

if(aux->ano<10){strcpy(v3,"000");}// se o ano é menor que 10 variavel de auxilio v3 recebe "000"
else{
    if(aux->ano<100){strcpy(v3,"00");}// se o ano é menor que 10 variavel de auxilio v3 recebe "00"
    else{
        if(aux->ano<1000){strcpy(v3,"0");}// se o ano é menor que 10 variavel de auxilio v3 recebe "0"
        else{ strcpy(v3,"");}// senao a variavel de auxilio v3 fica vazia
    }
}

printf(" DD/MM/AAAA \n");
if(flag==1){//se a flag recebida por parametro no chamamento da funcao for 1 imprimi  ...
       printf("   /  /%s%d \n",v3,aux->ano);

}else {
    if(flag==2){//se a flag recebida por parametro no chamamento da funcao for 2 imprimi  ...
    printf("   /%s%d/%s%d \n",v2,aux->mes,v3,aux->ano);
    }else {
        if (flag==3){//se a flag recebida por parametro no chamamento da funcao for 3 imprimi  ...
            printf(" %s%d/%s%d/%s%d \n",v1,aux->dia,v2,aux->mes,v3,aux->ano);
        }
    }
}
}

void vdata(Tdados voo[TAM],Tdados *aux){//captura a data validada em inteiro

int flagano,flagmes,flagdia,igualmes,flag;//flags de validacao
int limitedia;
int r4,r100,r400;
char resp;//reposta que chama usada para a pergunta se digitou corretamenete a hora
flagano=flagmes=flagdia=limitedia=igualmes=0;//inicializando variaveis

flag=0;
do{//REPITA enquanto ATE que ano seja MAIOR OU IGUAL que o ano ATUAL
    if(flag==2){//exibe a mensagem se nao passar na condicao
    printf("\n\n\n----------------------------------------------------------\n");
    printf(" ANO DIGITADO NÃO PODE SER MENOR QUE ANO ATUAL !!!\n");
    printf("----------------------------------------------------------\n");
    system("pause");//pausa o programa para permitir que o usuario leia o aviso
    }
    printf("Informe o ANO .: ");
    scanf("%d",&aux->ano);//leitura de dados
    fflush(stdin);//limpar buffer apos a leitura
    if(aux->ano>=aux->iano){//se ano fornecido for maior ou igual que o ano inicial
        if(aux->ano>9999){
            printf("-------------------------------------------------\n");
            printf("!!! O ANO DIGITADO NÃO ESTÁ NO FORMATO AAAA !!!\n");
            printf("!!! O ANO DEVE TER APENAS 4 DIGITOS !!!\n");
            printf("-------------------------------------------------\n");
            flag=3;
        }else{
            flag=1;//.. entao flag recebe 1
        }
    }else{// senao se o ano digitado por menor que o atual
        flag=2;//.. entao flag recebe 0
    }
}while(flag==2 || flag==3);//repete o laço enquanto flag for 2

if(aux->ano==aux->iano){//se ano digitado for IGUAL o ano ATUAL
    flagano=1;
}
flag=0;
do{//REPITA enquanto o Ano e mes for MENOR que o atual
    mostradigitadodata(aux,1);//mostra a data com apenas ano
    printf("Informe o MES .: ");
    scanf("%d",&aux->mes);//captura valor da variavel
    if(flagano==1){//se o ano digitado é igual ao  ano atual
       if(aux->mes>=aux->imes) {//se  o mes digitado for maior ou igual ao mes atual

            if(aux->mes>0 && aux->mes<=12){//se o mes digitado estiver no intervalo de 1 a 12
                flagmes=1;
            }else{//senao exibe uma mensagem ao usuario
                flagmes=0;
                flag=1;
                printf("\n\n\n----------------------------------------------------\n");
                printf("\ O MES DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A 12\n");
                printf("----------------------------------------------------\n");
                system("pause");//pausa o programa para permitir que o usuario leia a mensagem
            }

        }else{//se o ANO ATUAL for igual ao ano digitado e o MES ATUAL eh menor que o mes digitado
            flag=1;
            printf("\n\n\n----------------------------------------------------\n");
            printf("\ O MES DIGITADO É MENOR QUE O MES ATUAL\n");
            printf("----------------------------------------------------\n");

            if(!(aux->mes>0 && aux->mes<=12)){//se alem disso estiver fora do intervalo .. exibir a mensagem
                printf("\n\n\n\t\t E \n");
                printf("\n\n\n----------------------------------------------------\n");
                printf("\ O MES DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A 12\n");
                printf("----------------------------------------------------\n\n");
            }

            flagmes=0;
            system("pause");//pausa o programa para permitir que o usuario leia a mensagem
        }
    }else{//se o ano digitado nao for igual ao atual ...
        if ((flagano==0)
            &&(aux->mes>0 && aux->mes<=12)){
             flagmes=2;//se ano for maior verifica se o mes digitado esta entre 1 e 12
        }else{//se nao estiver no intervalo exibe uma mensagem

            printf("\n\n\n----------------------------------------------------\n");
            printf("\ O MES DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A 12\n");
            printf("----------------------------------------------------\n");
            system("pause");//pausa o programa para permitir que o usuario leia a mensagem
        }
    }

}while(!((flagmes==1) || (flagmes==2)));//repete ateh que flagmes seja igual a  1 ou 2

do{//REPITA ate que o dia digitado esteja entre o limite de dia que o mes digitado permite
mostradigitadodata(aux,2);//mostra data com mes e ano
printf("Informe o DIA .: ");
scanf("%d",&aux->dia);//captura o valor da variavel
    switch(aux->mes)
    {
        case 1 : case 3 :case 5 : case 7 : case 8 : case 10 : case 12 :
        limitedia=31;//caso o mes seja 1 3 5 7 8 10 12 o valor da variavel vira 31
            if(aux->dia>0 && aux->dia<=limitedia){//se o dia digitado estiver no intervalo
                flagdia=1;
            }else{//se nao estiver no intervalo possivel do mes entao exibe a mensagem
                flagdia=0;
                printf("\n\n\n----------------------------------------------------------------\n");
                printf("\ O DIA DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A %d\n",limitedia);
                printf("----------------------------------------------------------------\n");
                system("pause");//pausa o programa para permitir que o usuario leia a mensagem
            }
        break;

        case 4:case 6:case 9:case 11:
        limitedia=30;//caso o mes seja 4 6 9 11 o valor da variavel vira 30
            if(aux->dia>0 && aux->dia<=limitedia){
                flagdia=1;
            }else{//se nao estiver no intervalo possivel do mes entao exibe a mensagem
                flagdia=0;
                printf("\n\n\n----------------------------------------------------------------\n");
                printf("\ O DIA DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A %d\n",limitedia);
                printf("----------------------------------------------------------------\n");
                system("pause");//pausa o programa para permitir que o usuario leia a mensagem
            }
        break;

        case 2:
            r4=aux->ano % 4;
            r100=aux->ano % 100;
            r400=aux->ano % 400;
            if((r4==0)&&((r100!=0)||(r400==0))){
                limitedia=29;
                if(aux->dia>0 && aux->dia<=limitedia){//se o dia digitado estiver no intervalo
                    flagdia=1;
                }else{//se nao estiver no intervalo possivel do mes entao exibe a mensagem
                            printf("\n\n\n----------------------------------------------------------------\n");
                            printf("\ O DIA DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A %d\n",limitedia);
                            printf("----------------------------------------------------------------\n");
                            system("pause");//pausa o programa para permitir que o usuario leia a mensagem
                            flagdia=0;
                }

            }else {
                limitedia=28;
                    if(aux->dia>0 && aux->dia<=limitedia){//se o dia digitado estiver no intervalo
                        flagdia=1;
                    }else{//se nao estiver no intervalo possivel do mes entao exibe a mensagem
                        printf("\n\n\n----------------------------------------------------------------\n");
                        printf(" O DIA DIGITADO NÃO ESTÁ NO INTERVALO DE 1 A %d\n",limitedia);
                        printf("----------------------------------------------------------------\n");
                        system("pause");//pausa o programa para permitir que o usuario leia a mensagem
                        flagdia=0;
                    }
            }
            break;
    }//FIM DO SWITCH CASE
    if(aux->mes==aux->imes){//se mes digitado for IGUAL o mes ATUAL
    igualmes=1;
    }else{ igualmes=0;}
    //se o mes e ano for igual e o dia digitado eh menor que o dia atual exibe a mensagem
    if(aux->dia< aux->idia && flagano==1 && igualmes==1){
        printf("\n\n\n----------------------------------------------------------------\n");
        printf(" O DIA DIGITADO É MENOR QUE O DIA ATUAL \n");
        printf("----------------------------------------------------------------\n");
        system("pause");//pausa o programa para permitir que o usuario leia a mensagem
    flagdia=0;
    }
}while(!(flagdia==1));//se tiver entre o limite  nao repete
flag=0;
do{

    //MOSTRA DATA DIGITADA ATEH AGORA
    mostradigitadodata(aux,3);//mosta a data digitada

    if(flag==1){// mostra a mensagem se o valor for diferente de S e N
        printf("\n\n----------------------------------------------\n");
        printf("!!! VOCÊ DIGITOU UM VALOR DIFERENTE DE (S/N) !!!\n");
        printf("----------------------------------------------\n\n");
    }

    printf("digitou corretamente ?(S/N)");
    resp=getche();//capturar resposta
    resp=towupper(resp);//forcar caracter para maiusculo
    printf("\n\n");
```

```
    if(resp!='S' && resp!='N'){//se o valor for diferente de S e N entao repete
        flag=1;
    }else{//senao para o loop
        flag=0;
    }

}while(flag==1);//repete enquanto o valor de resp for diferente de S e N
if(resp=='N'){//se nao estiver correta a data repete o procedimento
vdata(voo,aux);//chama a funcao data novamente se for respondido N
}


}

void datatostring(Tdados voo[TAM],Tdados *aux){//converte os campos de data inteiro  e os joga para uma string
vdata(voo,aux);//chama funcao de ler data
char data[11],dia[3],mes[3],ano[5];
itoa(aux->dia, dia, 10);// converte inteiro dia para string dia
itoa(aux->mes, mes, 10);// converte inteiro mes para string mes
itoa(aux->ano, ano, 10);// converte inteiro ano para string ano
if(aux->dia<10){//se o dia for menor que DEZ atribui um digito ZERO a primeira posicao da string
    strcpy(aux->data,"0");//adiciona ZERO a primeira posicao da string data
    strcat(aux->data,dia);//concatena(soma, junta) a string data com a string dia
}
else{//senao copia normalmente o dia na sequencia dia/mes/ano
strcpy(aux->data,dia);//copia o valor da string dia para a string data
}
strcat(aux->data,"/");//concatena(soma, junta) a string data um caracter '/'

if(aux->mes<10){//se o dia for menor que DEZ atribui um digito ZERO a primeira posicao da string
strcat(aux->data,"0");//concatena(soma, junta) a string data o caracacter '0'
strcat(aux->data,mes);//concatena(soma, junta) a string data a string mes
}
else{//senao copia normalmente o mes na sequencia dia/mes/ano
strcat(aux->data,mes);//concatena(soma, junta) a string data a string mes
}
strcat(aux->data,"/");//concatena(soma, junta) a string data um caracter '/'

if(aux->ano<10){//se o dia for menor que DEZ atribui um digito ZERO a primeira posicao da string
strcat(aux->data,"000");//concatena(soma, junta) a string data os caracteres'000'
strcat(aux->data,ano);//concatena(soma, junta) a string data a string ano
}
else{
    if(aux->ano<100){//se o dia for menor que DEZ atribui um digito ZERO a primeira posicao da string
    strcat(aux->data,"00");//concatena(soma, junta) a string data os caracteres'00'
    strcat(aux->data,ano);//concatena(soma, junta) a string data a string ano
    }else{
        if(aux->ano<1000){//se o dia for menor que DEZ atribui um digito ZERO a primeira posicao da string
        strcpy(aux->data,"0");//concatena(soma, junta) a string data o caracter'0'
        strcat(aux->data,ano);
        }else{//senao copia normalmente o ano na sequencia dia/mes/ano
                strcat(aux->data,ano);//concatena(soma, junta) a string data a string ano
        }
        }
    }
}

void atualdata(Tdados voo[TAM],Tdados *aux){//captura a data atual para o sistema
vdata(voo,aux);//chamando a leitura de data
aux->idia=aux->dia;//joga o dia digitado para o dia inicial do programa
aux->imes=aux->mes;//joga o mes digitado para o mes inicial do programa
aux->iano=aux->ano;//joga o ano digitado para o ano inicial do programa

}

Tdados inserirpass(Tdados voo[TAM],Tdados *aux){//funcao usada no metododecadastro para inserir passageiros
    strcpy(aux->nome,validanome());//copia o retorno da funcao para variavel nome
    aux->origem=validaorigem();//atribui o retorno da funcao para variavel origem
    aux->destino=validadestino(aux->origem);//atribui o retorno da funca para variavel destino
    //strcpy(aux->data,validata());// atribui o retorno da funcao para a variavel data
    datatostring(voo,aux);//chama a funcao que converte os campos dia me e ano para uma unica string DD/MM/AAAA
return *aux;//retorno da funcao
}

Tdados buscalista(Tdados voo[TAM],Tdados *aux){//funcao usada em relotorio 3 e 4 para obter os dados principais de busca
    aux->origem=validaorigem();//atribui o retorno da funcao para variavel origem
    aux->destino=validadestino(aux->origem);//atribui o retorno da funca para variavel destino
    //strcpy(aux->data,validata());// atribui o retorno da funcao para a variavel data
    datatostring(voo,aux);//chama a funcao que converte os campos dia me e ano para uma unica string DD/MM/AAAA
return *aux;//retorno da funcao
}

char *excluirnome(char nome[50]){//funcao usada para excluir nome do passageiro
    strcpy(nome,"");//atribui vazio para a variavel
return nome;//retorno da funcao
}

void zeravetoraux(char vaux[16][50]){//procedimento para zerar vaux
    int i;
    for(i=0;i<16;i++){//varre o vetor de string e deixa todos campos vazios
    strcpy(vaux[i],"");//atribui vazio para a posicao do vetor
    }
}

void copiavetor(char lista[16][50]){//procedimento usado para jogar primeiro da fila de espera para lista de voo
    int i,j;//contadores
    char vaux[16][50];//variavel string
    zeravetoraux(vaux);//deixa vazio o vetor auxiliar antes de usar..
    j=0;//inicializando contador
    for(i=0;i<16;i++){//copia do vetor original para o auxiliar sem os vazios
        if(strcmp(lista[i],"")!=0){//se sao diferente de vazio
            strcpy(vaux[j],lista[i]);//copia o conteudo da posicao de lista para a posicao de vaux
            j++;//contador
        }
    }
    for(i=0;i<16;i++){//copia do vetor auxiliar para vetor original
    strcpy(lista[i],vaux[i]);//copia o conteudo da posicao de vaux devolta para a posicao de lista
    }
}

int getPosvoo(Tdados voo[TAM],Tdados *aux){//encontra a posicao do voo especifico ja cadastrado

int pos,i;
pos=-1;//atribuir uma posicao diferente das possiveis de um vetor para ter certeza que encontrou a posicao
i=0;//inicializa o contador

while(pos==-1 && i<TAM){//enquanto a possicao continuar -1 e o contador i for menor que o limite do vetor faça ...
  if(
    voo[i].origem==aux->origem &&
    voo[i].destino==aux->destino &&
    (strcmp(voo[i].data,aux->data)==0)){//se encontrar em alguma posicao do vetor origem e destino e data igual a digitada ...
    pos=i;//salva a posicao que foi encontrado os items iguais ao digitado

    }
    else{//senao continua o loop varrendo o vetor
        i++;//acrescenta +1 ao contador i
    }
}
//caso nao encontrar origem e destino e data iguais permanecera -1 para variavel pos
return pos;//retorna o valor da funcao
}

int getPosLivrepass(char lista[16][50]){// encontra a posicao livre para armazenar o passageiro no voo
int pos,i;//guardador de posicao ,contador
pos=-1;//atribuir uma posicao diferente das possiveis de um vetor para ter certeza que encontrou a posicao
i=0;//inicializa o contador i

while(pos==-1 && i<=4){//enquanto a possicao continuar -1 e o contador i for menor ou igual a 4 faça ...
  if(strlen(lista[i])==0){//se tiver alguma posicao do vetor que esteja vazia me diga
    pos=i;//armazena a posicao que foi encontrado vazio
    }
i++;//acrescenta mais 1 ao contador i
}
//caso nao encontrar origem e destino e data iguais permanecera -1 para variavel pos
return pos;//retorna o valor da funcao

}

int getPosLivrepass2(char lista[16][50]){// encontra a posicao livre para armazenar o passageiro na fila de espera
int pos,i;//guardador de posicao ,contador
pos=-1;//atribuir uma posicao diferente das possiveis de um vetor para ter certeza que encontrou a posicao
i=5;//inicializa o contador i

while(pos==-1 && i<=14){///enquanto a possicao continuar -1 e o contador i for menor ou igual a 14 faça ...
  if(strlen(lista[i])==0){//se tiver alguma posicao do vetor que esteja vazia me diga
    pos=1;//armazena a posicao que foi encontrado vazio
    }
i++;//acrescenta mais 1 ao contador i
}
//caso nao encontrar origem e destino e data iguais permanecera -1 para variavel pos
return pos;//retorna o valor da funcao

}

int getPosExcluido(char lista[16][50],Tdados *aux){//encontra a posicao que se encontra o nome do passageiro digitado num num voo ou fila de espera
int pos,i;//guardador de posicao ,contador
pos=-1;//atribuir uma posicao diferente das possiveis de um vetor para ter certeza que encontrou a posicao
i=0;//inicializa o contador i

while(pos==-1 && i<=14){//enquanto a possicao continuar -1 e o contador i for menor ou igual a 14 faça ...
  if(strcmp(lista[i],aux->nome)==0){//compara o nome digitado com os nomes armazenados na lista de voo e lista de espera
    pos=i;//o primeiro que encontrar com este nome me diga
    }
i++;//acrescenta +1 ao conta i
}
//caso nao encontrar origem e destino e data iguais permanecera -1 para variavel pos
return pos;// retorna a posicao a funcao

}

int getPosLivrevoo(Tdados voo[TAM],Tdados *aux){//encontra uma posicao no vetor para armazenar voo
int pos,i;//guardador de posicao ,contador
pos=-1;//atribuir uma posicao diferente das possiveis de um vetor para ter certeza que encontrou a posicao
i=0;//inicializa o contador i

while(pos==-1 && i<TAM){//enquanto a possicao continuar -1 e o contador i for menor que o limite do vetor faça ...
  if(
    voo[i].origem==0 &&
    voo[i].destino==0 &&
    (strcmp(voo[i].data,"")==0) ){//se encontrar em alguma posicao do vetor em que  origem e destino sejam 0 e data seja vazio me diga ...
    pos=i;//salva a posicao que foi encontrado os items iguais ao digitado

    }
    else{//senao continua o loop varrendo o vetor
        i++;//acrescenta +1 ao contador i
    }
}
//caso nao encontrar origem e destino e data iguais permanecera -1 para variavel pos
return pos;//retorna o valor da funcao
}

int getnumeropassvoo(char lista[16][50],Tdados *aux){//conta quantos passageiros tem na lista de voo
int i;//contador i
aux->contvoo=0;//inicializando contador

for(i=0;i<=4;i++){//varre a lista de voo e conta quantos nomes tem escrito
  if(strlen(lista[i])!=0){
    aux->contvoo++;//incrementa +1 ao contador de voo
  }
}
//caso nao encontrar o contador permanece 0
return aux->contvoo;//retorna valor da funcao
}

int getnumeropassvooespera(char lista[16][50],Tdados *aux){//conta quantos passageiros tem na lista de espera
int i;//contador i
aux->contespera=0;//inicializando contador

for(i=5;i<=14;i++){//varre a lista de espera e conta quantos nomes tem escrito
  if(strlen(lista[i])!=0){
    aux->contespera++;//incrementa +1 ao contador de voo
  }
```

```
}
//caso nao encontrar o contador permanece 0
return aux->contespera;//retorna valor da funcao
}

void msgsucesso(){//procedimento de mensagem de reserva de voo realizado com sucesso
    espacos();//insere enter na na tela
    printf("-----------------------------------------\n");
    printf(" Reserva de voo Realizada com Sucesso !!!\n");
    printf("-----------------------------------------\n");
    espacos();//insere enter na na tela
    system("pause");//pausa a tela para o usuario ver a mensagem
}

void msgfiladeespera(){//procedimento de mensagem lista de voo lotada , voce foi colocado na fila de espera
    espacos();//insere enter na na tela
    printf("-----------------------------------------\n");
    printf(" Lista de vôo Lotada !!!\n");
    printf(" Você foi colocado na Fila de Espera.\n");
    printf("-----------------------------------------\n");
    espacos();//insere enter na na tela
    system("pause");//pausa a tela para o usuario ver a mensagem
}

Tdados metodocadastro(Tdados voo[TAM],Tdados *aux){//funcao que controla o agendamento de voo
    system("cls");//limpa tela
int posvoocad,posvoo,pospass;
inserirpass(voo,aux);// captura nome, destino , origem e data
posvoocad=getPosvoo(voo,aux);//procura se o voo ja esta cadastado e retorna -1 senao tiver

if(posvoocad!=-1){//se o voo ta cadastrado
    pospass=getPosLivrepass(voo[posvoocad].lista);//procura posicao livre na lista de voo para atribuir o passageiro

    if(pospass!=-1){//se a posicao livre foi encontrada
    strcpy(voo[posvoocad].lista[pospass],aux->nome);//adiciona o passageiro a lista de voo
    msgsucesso();//exibir uma mensagem de sucesso no cadastro
    }
    else{//se nao encontro passageiro na lista de voo procura possicao livre na fila de espera
        pospass=getPosLivrepass2(voo[posvoocad].lista);// captura a posicao livre na fila de espera  para armazenar o nome
        if(pospass!=-1){// se a posicao livre para adicionar o passageiro foi encontrada entao ...
        msgfiladeespera();//exibir uma mensagem que foi jogado para fila de espera
        strcpy(voo[posvoocad].lista[pospass],aux->nome);//atribui o nome do passageiro a fila de espera na posicao encontrada..
        }
        else{//senao foi entrado nenhuma possicao nem na lista de voo nem na fila de espera entao ...
            espacos();// adicionar enter na tela
            printf("Lista de Vôo e Lista de Espera Lotadas !!!\n");
            printf("Favor escolher outra data.\n");
            system("pause");//pausa a tela para o usuario ver a mensagem
        }
    }
    voo[posvoocad].contvoo=getnumeropassvoo(voo[posvoocad].lista,aux);//obtem o numero atual de fila de voo
    voo[posvoocad].contespera=getnumeropassvooespera(voo[posvoocad].lista,aux);//obtem o numero atual de fila de espera
}
else{// se o voo nao esta cadastrado entao ....
    posvoo=getPosLivrevoo(voo,aux);// procurando a possicao de voo livre , se nao encontrar posicao retorna -1
    if(posvoo!=-1){//se encontrou uma posicao livre de voo para registrar  entao ...
    voo[posvoo].origem=aux->origem;//registra nessa possicao a origem
    voo[posvoo].destino=aux->destino;//registra nessa possicao o destino
    strcpy(voo[posvoo].data,aux->data);//registra nessa possicao a data
    voo[posvoo].ano=aux->ano;//registra nessa possicao o ano
    voo[posvoo].mes=aux->mes;//registra nessa possicao o mes
    voo[posvoo].dia=aux->dia;//registra nessa possicao o dia
    strcpy(voo[posvoo].lista[0],aux->nome);//registra nessa possicao o nome

    voo[posvoo].contvoo=getnumeropassvoo(voo[posvoo].lista,aux);//obtem o numero atual de fila de voo
    voo[posvoo].contespera=getnumeropassvooespera(voo[posvoo].lista,aux);//obtem o numero atual de fila de espera
    msgsucesso();//exibir uma mensagem de sucesso no cadastro
    }
    else{//se nao encontrar possicao de voo é porque acabou o limite de registro para cadastros
        espacos();//adiciona enter na tela
        printf("Impossivel Cadastrar mais Vôos ao Programa os registros estao Lotados !!! \n");
        system("pause");// pausa a tela para o usuario ver a mensagem
    }
}

return *aux;//retorna o valor da funcao
}

void cabecalhorelatorio(char palavra[20],Tdados *aux){//procedimento de cabecalho para o relatorio 3 e 4

    printf(" Lista %s \n",palavra);
    printf("------------------------------\n");
    printf(" Origem ..: %s\n",v[aux->origem]);
    printf(" Destino .: %s\n",v[aux->destino]);
    printf(" Data ...: %s\n",aux->data);
    printf("------------------------------\n");
}

void cabecalhorelatorio1(char palavra[20],Tdados voo[TAM],Tdados *aux,int total){//procedimento de cabecalho para o relatorio 1
int i;
    if(total==-1){// nao axou possicao livre com getposslivrevoo é sinal de que todas todos registros estao lotados entao imprimi tudo..
        total=TAM;
    }

    printf(" Lista de %s\n",palavra);
    printf("-------------------------------------------------------------------\n");
    for(i=0;i<total;i++){//imprime todos voos cadastrados no sistema...
    printf(" Origem ..............................: %s\n",v[voo[i].origem]);
    printf(" Destino .............................: %s\n",v[voo[i].destino]);
    printf(" Data ................................: %s\n",voo[i].data);
    printf(" N° de passageiros ...................: %d\n",voo[i].contvoo);
    printf(" N° de passageiros na fila de espera ..: %d\n",voo[i].contespera);
    printf("-------------------------------------------------------------------\n");
    }
    if(i==0){//se  o indice i do for continuar zero é por que nao tem nenhum voo criados ainda..
    espacos();//insere enter na tela
    printf("-----------------------------------------\n");
    printf(" NO MOMENTO NÃO EXISTEM VÔOS CRIADOS !!! \n");
    printf("-----------------------------------------\n");
    espacos();//insere enter na tela
    }


}

void cabecalhorelatorio2(char palavra[20],Tdados voo[TAM],Tdados *aux,int total){//procedimento  de cabecalho para o relatorio 2
int i,flag;//variaveis
flag=0;//atribuindo zero a flag
    if(total==-1){// nao axou possicao livre com getposslivrevoo é sinal de que todas todos registros estao lotados entao imprimi tudo..
        total=TAM;
    }

    printf(" Lista de %s\n",palavra);
    printf("-------------------------------------------------------------------\n");
    for(i=0;i<total;i++){//imprimi todos voo cadastrados porem ...
    if(voo[i].contvoo>=2){//..so imprime se for MAIOR ou IGUAL a 2
    printf(" Origem ..............................: %s\n",v[voo[i].origem]);
    printf(" Destino .............................: %s\n",v[voo[i].destino]);
    printf(" Data ................................: %s\n",voo[i].data);
    printf(" N° de passageiros ...................: %d\n",voo[i].contvoo);
    printf(" N° de passageiros na fila de espera ..: %d\n",voo[i].contespera);
    printf("-------------------------------------------------------------------\n");
    flag=1;//atribui 1 a variavel para evitar que imprima o proximo if
    }

    }
    if(flag==0){//se  o indice i do for continuar zero é por que nao tem nenhum voo criados ainda..
    espacos();//inserir enter na tela
    printf("-----------------------------------------------\n");
    printf(" NO MOMENTO NÃO EXISTEM VÔOS CONFIRMADOS !!! \n");
    printf("-----------------------------------------------\n");
    espacos();//inserir enter na tela
    }
}

void exibe_r1(Tdados voo[TAM],Tdados *aux){//procedimento de exibicao de relatorio 1
system("cls");//limpa a tela
int total;//variavel de limite para usar na no procedimento de cabecalho
total=getPosLivrevoo(voo,aux);//captura uma posicao livre de voo
cabecalhorelatorio1("Vôos Criados",voo,aux,total);//chama o procedimento de relatorio enviando total por parametro

printf("\n\n\n\n\n\n\n\n\n\n\n\n");//imprime alguns enter na tela
}

void exibe_r2(Tdados voo[TAM],Tdados *aux){//procedimento de exibicao de relatorio 2
system("cls");//limpa tela
int total;//variavel de limite para usar na no procedimento de cabecalho
total=getPosLivrevoo(voo,aux);//captura uma posicao livre de voo
cabecalhorelatorio2("Vôos Confirmados",voo,aux,total);//chama o procedimento de relatorio enviando total por parametro

printf("\n\n\n\n\n\n\n\n\n\n\n\n");//imprime alguns enter na tela
}

void exibe_r3(char var[16][50],Tdados *aux){//sub procedimento de exibicao do relatorio 3
    system("cls");//limpa tela
    int i,li,lf,flag;
    flag=0;
    li=0; lf=4;//atribuindo limite incial e final do for

    cabecalhorelatorio("de Vôo",aux);//exibe o cabecalho do relatorio de 3

    for(i=li;i<=lf;i++){//varre a lista de voo de zero a 4
        if(!(strcmp(var[i],""))==0){//se for diferente de vazio imprime
        printf("%d". %s \n",i+1,var[i]);//deve se colocar +1 no  i pois o vetor comeca em zero
        flag=1;//se a condicao de if passar apenas uma vez o flag sinaliza 1
        }
    }

    if(flag==0){//se nao for encontrado nenhum vazio no bloco for anterior flag continua ZERO entao imprimi ....
    printf("--------------------------------\n");
    printf(" VÔO SEM PASSAGEIROS !!!\n");
    printf("--------------------------------\n");
    }

    printf("\n\n\n\n\n\n\n\n\n\n\n\n");// inserir enter na tela
}

void exibe_r3_lista(Tdados voo[TAM],Tdados *aux){//procedimento de exibicao do relatorio 3
    int indice;

    buscalista(voo,aux);//captura nome, origem , destino e data
    indice=getPosvoo(voo,aux);//recebe da funcao a posicao do registro que foi encontrado o voo pedido
    if(indice!=-1){// se encontra o voo ...
    exibe_r3(voo[indice].lista,aux);//exibe a lista de voo
    }

    else{//senao encontrar exibe a mensagem ....
    printf("-------------------------------------\n");
    printf(" VÔO NÃO CADASTRADO !!!\n");
    printf("-------------------------------------\n");
    }

}

void exibe_r4(char var[16][50],Tdados *aux){//sub procedimento de exibicao do relatorio 4
    system("cls");//limpa a tela
    int i,li,lf,flag;
    flag=0;
    li=5; lf=14;//atribui o limite inicial e final para os controladores do for

    cabecalhorelatorio("da fila de Espera",aux);//exibe o relatorio do cabecalho 3


    for(i=li;i<=lf;i++){//varre a lista de espera de 5  a 14
        if(!(strcmp(var[i],""))==0){//imprime todos nomes que nao estiverem vazios
        printf("%d". %s \n",i-5+1,var[i]);
        flag=1;//se encontrar vazio atribui 1 a flag
        }
    }

    if(flag==0){//se nao foi encontrado nenhum nome no bloco anterior do for entao exibe
    printf("--------------------------------\n");
    printf(" FILA DE ESPERA VAZIA !!!\n");
```

```
    printf("--------------------------------\n");
    }

    printf("\n\n\n\n\n\n\n\n\n\n\n\n");//inserir enter na tela
}

void exibe_r4_lista(Tdados voo[TAM],Tdados *aux){//procedimento de exibicao de relatorio 4
    int indice;

    buscalista(voo,aux);//captura nome , origem, destino , data
    indice=getPosvoo(voo,aux);//atribui a variavel indice a posicao do vetor para o voo requisitado
    if(indice!=-1){//se o voo for encontrado
    exibe r4(voo[indice].lista,aux);//exibe a lista de espera do voo especificado
    }
    else{//senao for encontrado exibe a mensagem ...
    printf("-----------------------------------------\n");
    printf(" VÔO NÃO CADASTRADO !!!\n");
    printf("-----------------------------------------\n");
    }
}

void metodoexclusao(Tdados voo[TAM],Tdados *aux){//procedimento de exclusao
char flag[15];
int posvoocad,pe;//posicao voo e posicao exclusao

inserirpass(voo,aux);//capturar nome, origem , destino , data
posvoocad=getPosvoo(voo,aux);//encontrar a posicao do registro que encontra esse passageiro
pe=getPosExcluido(voo[posvoocad].lista,aux);//encontrar a possicao da lista que encontra esse passageiro

if(pe!=-1){//encontrou a posicao na lista de passageiros
    excluirnome(voo[posvoocad].lista[pe]);//excluir o passageiro
    copiavetor(voo[posvoocad].lista);//copia a lista original para um segundo vetor ignorando campos vazios e devolve a lista original por referencia

    if(pe>=0 && pe<=4){strcpy(flag,"fila de voo");}//se a possicao que foi encontrada estiver entre 0 e 4  entao  foi excluido do voo
    else{strcpy(flag,"Fila de Espera");}//se a possicao que foi encontrada estiver entre 5 e 14  entao  foi excluido da fila de espero

    voo[posvoocad].contvoo=getnumeropassvoo(voo[posvoocad].lista,aux);//obtem o numero de passageiros atual de fila de voo
    voo[posvoocad].contespera=getnumeropassvooespera(voo[posvoocad].lista,aux);//obtem o numero de passageiros atual de fila de espera
    espacos();//inserir enter na tela
    printf("-------------------------------------------------------------------------\n");
    printf("Passageiro encontrado na %s foi Excluido com Sucesso !!!\n",flag);
    printf("-------------------------------------------------------------------------\n");
    espacos();//inserir enter na tela
    system("pause");//pausar a tela para permitir que o usuario leia
}else{//se nao encontrou o nome do passageiro fornecido entao diga ...
    espacos();//inserir enter na tela
    printf("-------------------------------------------------------------------------\n");
    printf("Não foi possivel cancelar pois ,\n");
    printf("o usuario informado não foi encontrado nessa origem ,destino e data !!!\n");
    printf("-------------------------------------------------------------------------\n");
    espacos();//inserir enter na tela
    system("pause");//pausar a tela para permitir que o usuario leia
}

}

void Exibedatahoje(Tdados *aux){//procedimento para exibir a data do sistema
    char v1[2],v2[2],v3[4];// variaveis ajudantes
system("cls");//limpa a tela
espacos();//inserir enter na tela

if(aux->idia<10){ strcpy(v1,"0");  }//se o dia for menor que 10 entao copia 0 para a variavel ajudante
else{ strcpy(v1,"");}//senao deixa vazio

if(aux->imes<10){strcpy(v2,"0");}//se o mes for menor que 10 entao copia 0 para a variavel ajudante
else{ strcpy(v2,"");}//senao deixa vazio

if(aux->iano<10){strcpy(v3,"000");}//se o ano for menor que 10 entao copia 000 para a variavel ajudante
else{
    if(aux->iano<100){strcpy(v3,"00");}//se o ano for menor que 100 entao copia 00 para a variavel ajudante
    else{
        if(aux->iano<1000){strcpy(v3,"0");}//se o ano for menor que 1000 entao copia 0 para a variavel ajudante
        else{ strcpy(v3,"");}//senao deixa vazio
    }
}
//utiliza-se a variaveis ajudante na frente dos valores reais para autocompletar com ZERO quando for necessario
printf("-------------------------------------------------------\n");
printf(" DATA DO SISTEMA ..: ");
printf(" %s%d/%s%d/%s%d \n",v1,aux->idia,v2,aux->imes,v3,aux->iano);
printf("-------------------------------------------------------\n");
}

void logotipo(Tdados voo[TAM],Tdados *aux){ // logotipo
    Exibedatahoje(aux);// exibe a data do sistema
    espacos();//pular linhas
    printf("-------------------------------------------------------\n");
    printf("------------------- SEJA BEM VINDO -------------------\n");
    printf("-------------------------- A ----------------------------\n");
    printf("------- VOE FÁCIL LINHAS AÉREAS INTELIGENTES ---------\n");
    printf("-------------------------------------------------------\n\n");


}

int pergunta(){//funcao de pergunta do menu principal
int select;
    printf("Escolha uma das opções acima: ");
    scanf("%d",&select);
    fflush(stdin);//limpar buffer de entrada
    espacos();//inserir enter na tela
return select;//retorno da funcao
}

int m1(Tdados voo[TAM],Tdados *aux){//FUNCAO DE EXIBICAO DO MENU PRINCIPAL
    int select;
    system("cls");//limpar tela
    logotipo(voo,aux);//mostra logotipo
    printf("-------------------------------------------------------\n");
    printf("-------------------- MENU PRINCIPAL -------------------\n");
    printf("-------------------------------------------------------\n");
    printf("1 - Agendar vôo. \n");
    printf("2 - Cancelar passageiro. \n");
    printf("3 - Lista de relatórios. \n");
    printf("0 - Sair. \n");
    printf("-------------------------------------------------------\n");
    select=pergunta();//atribui o retor da funcao a variavel
    return select;//retorna o valor da funcao
}

int m2 (){//FUNCAO DE EXIBICAO DO MENU RELATORIOS
    int select;
    espacos();//inser enter na tela
    espacos();//inser enter na tela
    printf("-------------------------------------------------------\n");
    printf("------------------- LISTA DE RELATORIOS --------------\n");
    printf("-------------------------------------------------------\n");
    printf("1 - Lista de vôos criados. \n");
    printf("2 - Lista de vôos Confirmados. \n");
    printf("3 - Lista dos passageiros de um determinado vôo. \n");
    printf("4 - Lista da espera de um determinado vôo. \n");
    printf("0 - Voltar ao menu anterior. \n");
    printf("-------------------------------------------------------\n");
    select=pergunta();//atribui a variavel o retorno da funcao chamada
    return select;//retorno da funcao
}

int menu(Tdados voo[TAM],Tdados *aux){//funcao seletiva do menu principal
int select;

    do{
        select=m1(voo,aux);//atribui o retorno da funcao de menu para a variavel
    }while(!(select>=0 && select<=3));// repete enquanto nao estiver no intervalo de 0 a 3

    switch (select){
        case 1:
            metodocadastro(voo,aux);// Chamando a lista de marcacao voo
            break;//interrompendo o switch
        case 2:
            metodoexclusao(voo,aux);// chamando a cancelamento passageiro
            break;//interrompendo o switch
        case 3:
            relatorios(voo,aux);// chamando a lista de relatorio
            break;//interrompendo o switch
        case 0:
            exit(0);  // Fechar do programa (ignorando tudo)
            break;//interrompendo o switch
    }
    return select;//retorno da funcao
}

void relatorios(Tdados voo[TAM],Tdados *aux){
int select;

    do{
        select=m2();//atribui o retorno da funcao
    }while(!(select>=0 && select<=4));// repete ateh que numero do relatorio desejado esteja no intervalo de 0 a 4

     switch (select){
        case 1:
            exibe r1(voo,aux);// Chamando a lista de Voos Criados
            system("pause");//pausar a tela para permitir que o usuario leia
            break;//interrompendo o switch
        case 2:
            exibe r2(voo,aux);// Chamando a lista de Voos Confirmados
            system("pause");//pausar a tela para permitir que o usuario leia
            break;//interrompendo o switch
        case 3:
            exibe r3 lista(voo,aux);// Chamando a lista de passageiros de um determinado Voo
            system("pause");//pausar a tela para permitir que o usuario leia
            break;//interrompendo o switch
        case 4:
            exibe_r4_lista(voo,aux);// Chamando a lista de Espera de um determinado Voo
            system("pause");//pausar a tela para permitir que o usuario leia
            break;
        case 0:
            menu(voo,aux); // voltar para menu principal
            break;//interrompendo o switch
    }

}
```