**Problem 1:**

#include<stdio.h>

#include<string.h>

char names[100][30],id[100][9];

int n=0;

int\*pn;

char\* lTrim(char s[]){

int i = 0;

while (s[i] == ' ') i++;

if (i > 0) strcpy(&s[0], &s[i]);

return s;

}

char\* rTrim(char s[]){

int i = strlen(s)-1;

while (s[i] == ' ') i--;

s[i+1] = '\0';

return s;

}

char\* trim(char s[]){

rTrim(lTrim(s));

char \*ptr = strstr(s, " ");

while (ptr != NULL){

strcpy(ptr, ptr+1);

ptr = strstr(s, " ");

}

return s;

}

char\* nameStr (char s[]){

trim(s);

strlwr(s);

int L = strlen(s);

int i;

for (i = 0; i < L; i++)

if (i == 0 || (i > 0 && s[i-1] == ' '))

s[i] = toupper(s[i]);

return s;

}

void Add(char id[][9],char names[][30],int\*pn){

double t;

char ts[30];

int i , ex;

if ( n >= 100 ) printf("Fully list");

if ( n < 100 )

{

do{

printf("Student's ID' : ");

fflush(stdin);

scanf("%s",ts);

ex=0;

for( i = 0; i < \*pn; i++){

if (strcmp(ts,id[i]) == 0){

printf("The ID is existed");

ex=1;

}

}

}

while (ex==1);

strcpy(id[\*pn],ts);

printf("\nStudent's name' :");

scanf("%s",ts);

strupr(ts);

trim(ts);

strcpy(names[\*pn],ts);

(\*pn)++;

}

}

void Remove(char id[][9],char names[][30], int\*pn){

int i,j;

char ts[30];

printf("Enter ID to remove :");

scanf("%s",ts);

for (i=0;i<\*pn;i++){

if(strcmp(ts,id[i]) == 0) {

for(j = i+1; j < \*pn; j++){

strcpy(names[j-1], names[j]);

strcpy(id[j-1], id[j]);

}

(\*pn)--;

printf("OK Done\n");

}

}

}

void Find(char id[][9],char names[][30], int\*pn){

int i;

char ts[30];

printf("Student's name' :");

fflush(stdin);

scanf("%s",ts);

strupr(ts);

trim(ts);

for (i=0;i<n;i++){

if(strcmp(ts,names[i]) == 0){

printf("\nThe ID is : %s",id[i]);

}

}

}

void print(char id[][9],char names[][30], int\*pn){

int i, j;

for (i = 0; i < \*pn-1; i++)

for (j = \*pn-1; j > i; j--){

if (strcmp(id[j],id[j-1]) < 0){

char t[30];

strcpy(t,names[j]);

strcpy(names[j], names[j-1]);

strcpy(names[j-1], t);

char tu[9];

strcpy(tu,id[j]);

strcpy(id[j], id[j-1]);

strcpy(id[j-1], tu);

}

}

for (i = 0; i < (\*pn); i++) printf(" Name : %s , ID : %s\n ",names[i],id[i]);

}

int GetUserChoice(){

int choice;

printf(" MENU ");

printf("\n1 - Add a student");

printf("\n2 - Search a student");

printf("\n3 - Remove a student");

printf("\n4 - Sorting");

printf("\nOthers - Quit");

printf("\nSelect One: ");

scanf("%d%\*c",&choice);

return choice;

}

int main(){

int UserChoice;

do{

UserChoice=GetUserChoice();

switch(UserChoice){

case 1 :Add(id, names, &n);

break;

case 2 :Find(id, names, &n);

break;

case 3 :Remove(id, names, &n);

break;

case 4 :print(id, names, &n);

break;

default:printf("Gud day<3");

}

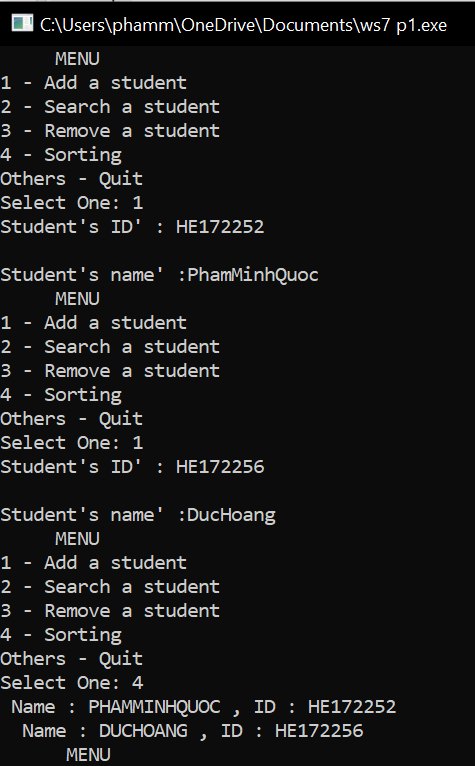
} while (UserChoice>0 && UserChoice<5);

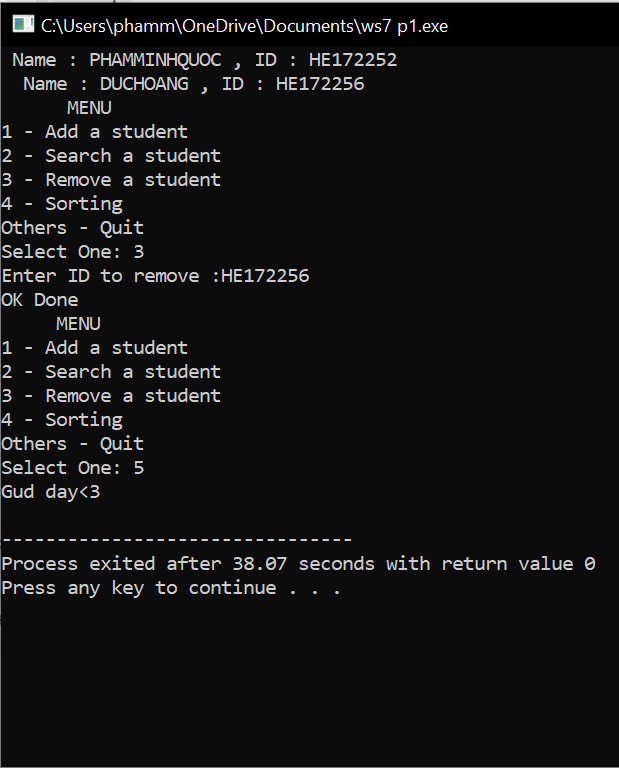
fflush(stdin);

getchar();

return 0;

}





**Problem 2:**

#include<stdio.h>

#include<string.h>

#include<conio.h>

char codes[100][9],names[100][21];

double salaries[100],allowances[100];

int n=0;

int\*pn;

void Add(char codes[][9],char names[][21], double salaries[], double allowances[], int\*pn)

{

double t;

char ts[21];

int i , ex;

if ( n >= 100 ) printf("Fully list");

if ( n < 100 ) {

do

{

printf("Employee's ID' : ");

fflush(stdin);

scanf("%s",ts);

ex=0;

for ( i = 0; i < \*pn; i++) {

if (strcmp(ts,codes[i]) == 0) {

printf("The ID is existed");

ex=1;

}

}

}while (ex==1);

strcpy(codes[\*pn],ts);

printf("\nEnter the name of employee :");

scanf("%s",ts);

strcpy(names[\*pn],ts);

printf("\nEnter the salary :");

scanf("%lf",&t);

salaries[\*pn]=t;

printf("\nEnter the allowances :");

scanf("%lf",&t);

allowances[\*pn]=t;

(\*pn)++;

}

}

void Find(char codes[][9],char names[][21], double salaries[], double allowances[], int\*pn)

{

int i;

char ts[21];

printf("Enter the name :");

fflush(stdin);

scanf("%s",ts);

for (i=0;i<n;i++){

if (strcmp(ts,names[i]) == 0) {

printf("\nThe ID is %s:",codes[i]);

printf("\nThe Salary is %lf:",salaries[i]);

printf("\nThe Allowances is %lf:",allowances[i]);

}

}

}

void Remove(char codes[][9],char names[][21], double salaries[], double allowances[], int\*pn){

int i,j;

char ts[21];

printf("Enter the ID to remove :");

scanf("%s",ts);

for (i=0;i<\*pn;i++){

if (strcmp(ts,codes[i]) == 0) {

for (j = i+1; j < \*pn; j++) {

strcpy(names[j-1], names[j]);

strcpy(codes[j-1], codes[j]);

salaries[j-1] = salaries[j];

allowances[j-1] = allowances[j];

}

(\*pn)--;

printf("OK Done!\n");

}

}

}

void print(char codes[][9],char names[][21], double salaries[], double allowances[], int\*pn)

{

int i, j;

for (i = 0; i < \*pn-1; i++)

for (j = \*pn-1; j > i; j--)

if ( (salaries[j] + allowances[j]) > (salaries[j-1] + allowances[j-1]) ) {

char t[21];

strcpy(t,names[j]);

strcpy(names[j], names[j-1]);

strcpy(names[j-1], t);

char tu[9];

strcpy(tu,codes[j]);

strcpy(codes[j], codes[j-1]);

strcpy(codes[j-1], tu);

double tg = salaries[j];

salaries[j]=salaries[j-1];

salaries[j-1]=tg;

double tb = allowances[j];

allowances[j]=allowances[j-1];

allowances[j-1]=tb;

}

for (i = 0; i < \*pn; i++)

printf("Code:%9s|Name:%21s|salary:%.4lf|allowance:%.4lf \n", codes[i], names[i], salaries[i],

allowances[i]);

if (\*pn)printf("Success!\n\n");

else printf("Nothing to print!\n\n");

}

int GetUserChoice()

{

int choice;

printf(" MENU ");

printf("\n1 - Add new employee");

printf("\n2 - Search");

printf("\n3 - Remove");

printf("\n4 - Sort");

printf("\nOthers - Quit");

printf("\nChoose: ");

scanf("%d%\*c",&choice);

return choice;

}

int main(){

int UserChoice;

do

{

UserChoice=GetUserChoice();

switch(UserChoice)

{

case 1 :Add(codes, names, salaries, allowances, &n);

break;

case 2 :Find(codes, names, salaries, allowances, &n);

break;

case 3 :Remove(codes, names, salaries, allowances, &n);

break;

case 4 :print(codes, names, salaries, allowances, &n);

break;

default:printf("OK BYE");

}

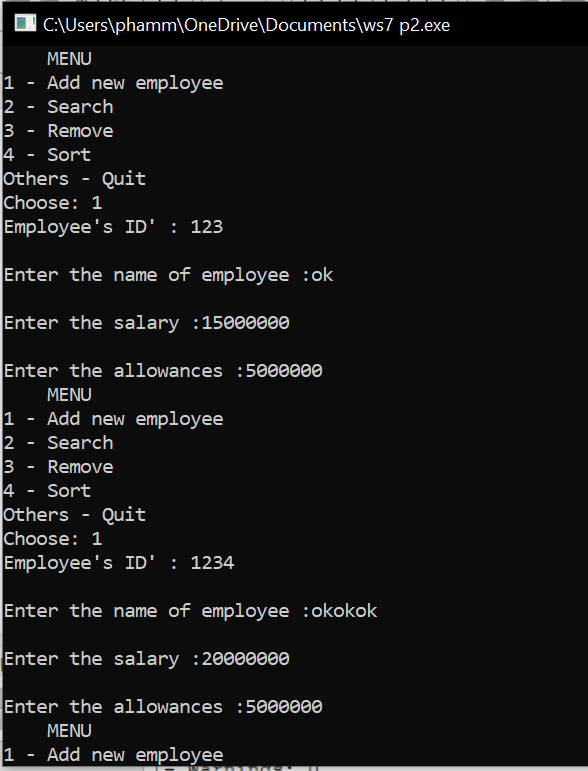
}while (UserChoice>0 && UserChoice<5);

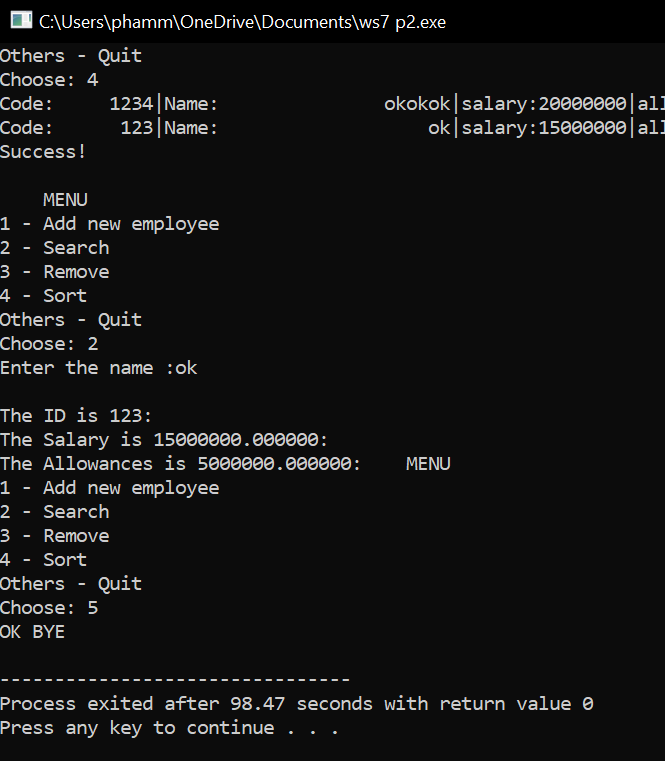
fflush(stdin);

getchar();

return 0;

}





**Problem 3:**

#include<stdio.h>

#include<string.h>

#include<conio.h>

char make[100][20],name[100][20];

int volume[100],duration[100],price[100];

int n=0;

char\* lTrim(char s[])

{

int i = 0;

while (s[i] == ' ') i++;

if (i > 0) strcpy(&s[0], &s[i]);

return s;

}

char\* rTrim(char s[])

{

int i = strlen(s)-1;

while (s[i] == ' ') i--;

s[i+1] = '\0';

return s;

}

char\* trim(char s[])

{

rTrim(lTrim(s));

char \*ptr = strstr(s, " ");

while (ptr != NULL)

{

strcpy(ptr, ptr+1);

ptr = strstr(s, " ");

}

return s;

}

char\* nameStr (char s[])

{

trim(s);

strlwr(s);

int L = strlen(s);

int i;

for (i = 0; i < L; i++)

if (i == 0 || (i > 0 && s[i-1] == ' '))

s[i] = toupper(s[i]);

return s;

}

void Add(char name[][20], char make[][20], int volume[], int price[], int duration[], int \*pn)

{

double t;

char ts[20];

int i , ex,volumes, prices, durations;

do

{

printf("Enter the name of drink : ");

fflush(stdin);

scanf("%s",ts);

ex=0;

for ( i = 0; i < \*pn; i++)

{

if (strcmp(ts,name[i]) == 0)

{

printf("The drink is existed");

ex=1;

}

}

}while (ex==1);

nameStr(ts);

strcpy(name[\*pn],ts);

printf("\nMade in : ");

scanf("%s",ts);

nameStr(ts);

strcpy(make[\*pn],ts);

printf("\nEnter the volume : ");

scanf("%d",&volumes);

printf("\nEnter the duration : ");

scanf("%d",&durations);

printf("\nEnter the price : ");

scanf("%d",&prices);

volume[\*pn] = volumes;

price[\*pn] = prices;

duration[\*pn] = durations;

(\*pn)++;

}

void printBaseMake(char name[][20], char make[][20], int volume[], int price[], int duration[], int n)

{

char makes[20];

printf("Enter the location : ");

scanf("%s", makes);

fflush(stdin);

nameStr(makes);

int i,check=0;

for (i = 0; i < n; i++)

{

if (strcmp(makes, make[i]) == 0)

{

printf("\n>>List of Soft Drink base on %s:\n", make[i]);

printf("Name : %s ; Make in : %s ; volume : %d ; price : %d ; duration : %d\n",name[i],make[i],volume[i],price[i],duration[i]);

check=1;

}

if (check==0) printf("\nThere are not any soft drinks on the list based on %s!!!\n", makes);

}

}

void printBaseVol(char name[][20], char make[][20], int volume[], int price[], int duration[], int n)

{

int v1,v2,i;

printf("Enter the min Volume : ");

scanf("%d",&v1);

printf("Enter the max Volume : ");

scanf("%d",&v2);

for (i=0;i<n;i++)

{

if (volume[i] >= v1 && volume[i] <= v2)

{

printf("Name : %s ; Make in : %s ; volume : %d ; price : %d ; duration : %d\n",name[i],make[i],volume[i],price[i],duration[i]);

}

}

}

void printAsc(char name[][20], char make[][20], int volume[], int price[], int duration[], int \*pn)

{

int i, j;

for (i = 0; i < (\*pn)-1; i++)

{

for (j = (\*pn)-1; j > i; j--)

{

if (price[j]+duration[j] < price[j-1]+duration[j-1])

{

char transName[20];

strcpy(transName, name[j-1]);

strcpy(name[j-1], name[j]);

strcpy(name[j], transName);

char transMake[20];

strcpy(transMake, make[j-1]);

strcpy(make[j-1], make[j]);

strcpy(make[j], transMake);

int transVol;

transVol = volume[j-1];

volume[j-1] = volume[j];

volume[j] = transVol;

int transPri;

transPri = price[j-1];

price[j-1] = price[j];

price[j] = transPri;

int transDur;

transDur = duration[j-1];

duration[j-1] = duration[j];

duration[j] = transDur;

}

}

}

for (i = 0; i < \*pn; i++)

printf("Name : %s ; Make in : %s ; volume : %d ; price : %d ; duration : %d\n",name[i],make[i],volume[i],price[i],duration[i]);

}

int GetUserChoice()

{

int choice;

printf("\n1 - Add new soft drink");

printf("\n2 - Printing out items which belong to a known make");

printf("\n3 - Printing out items whose volumes are between v1 and v2");

printf("\n4 - Sort");

printf("\nOthers - Quit");

printf("\nChoose: ");

scanf("%d%\*c",&choice);

return choice;

}

int main()

{

int UserChoice;

do

{

UserChoice=GetUserChoice();

switch(UserChoice)

{

case 1 :Add(name,make,volume,price,duration, &n);

break;

case 2 :printBaseMake(name,make,volume,price,duration, n);

break;

case 3 :printBaseVol(name,make,volume,price,duration, n);

break;

case 4 :printAsc(name,make,volume,price,duration, &n);

break;

default:printf("Bye !!!");

}

}while (UserChoice>0 && UserChoice<5);

fflush(stdin);

getchar();

return 0;

}

