**Final Exam**

**BIT 205**

“C++”



**Name: I Kadek Yau Dwi Mega Saputra**

**Stundent ID: E1800175**

**CODE C++:**

#include <iostream>

#include <fstream>

#include <conio.h>

#include <string.h>

#include <stdlib.h>

#include <ctime>

using namespace std;

typedef struct user

{

int id;

char Username[20], Email[50], DateJoined[25];

} user;

typedef struct member

{

int id;

char Level;

} member;

typedef struct trainer

{

int id;

char Specialty[100];

} trainer;

ifstream fi;

ofstream fo;

user usr;

member mbr;

trainer tnr;

char choose\_menu, choose\_level, info, r;

char username[20], email[50], datejoined[25], level, specialty[100];

int no,code;

void create(){

fo.open("usr.dat", ios::binary | ios::app);

//add User

usr.id=no;

cout << "Username : ";

cin >> usr.Username;

cout << "Email : ";

cin >> usr.Email;

cout << "Date Join : ";

cin >> usr.DateJoined;

cout <<endl;

cout << "=============Level=============" <<endl;

cout << "|B / b| To Become Beginner" <<endl;

cout << "|I / i| To Become Intermediate" <<endl;

cout << "|E / e| To Become Expert" <<endl;

cout <<"Select :";

cin>>choose\_level;

if(choose\_level=='B' or choose\_level=='b'){

fo.write((char \*) &usr, sizeof(usr));

fo.close();

//

cout << "Beginner" <<endl;

//add level (input level)

fo.open("member.dat", ios::binary | ios::app);

mbr.id=no;

mbr.Level=level;

fo.write((char \*) &mbr, sizeof(mbr));

fo.close();

//

cout << "===========Specialty===========" <<endl;

//add specialty

fstream f;

f.open("specialty.dat", ios::binary | ios::app);

tnr.id=no;

cout << "Specialty : ";

cin >> tnr.Specialty;

f.write((char \*) &tnr, sizeof(tnr));

f.close();

}

else if(choose\_level=='I' or choose\_level=='i'){

fo.write((char \*) &usr, sizeof(usr));

fo.close();

//

cout << "Beginner" <<endl;

//add level (input level)

fo.open("member.dat", ios::binary | ios::app);

mbr.id=no;

mbr.Level=level;

fo.write((char \*) &mbr, sizeof(mbr));

fo.close();

//

cout << "===========Specialty===========" <<endl;

//add specialty

fstream f;

f.open("specialty.dat", ios::binary | ios::app);

tnr.id=no;

cout << "Specialty : ";

cin >> tnr.Specialty;

f.write((char \*) &tnr, sizeof(tnr));

f.close();

}

else if(choose\_level=='E' or choose\_level=='e'){

fo.write((char \*) &usr, sizeof(usr));

fo.close();

//

cout << "Beginner" <<endl;

//add level (input level)

fo.open("member.dat", ios::binary | ios::app);

mbr.id=no;

mbr.Level=level;

fo.write((char \*) &mbr, sizeof(mbr));

fo.close();

//

cout << "===========Specialty===========" <<endl;

//add specialty

fstream f;

f.open("specialty.dat", ios::binary | ios::app);

tnr.id=no;

cout << "Specialty : ";

cin >> tnr.Specialty;

f.write((char \*) &tnr, sizeof(tnr));

f.close();

}

else{

cout<<"No Level Selected"<<endl;

}

}

void deletedata(){

cout << "=============Deleted=============" <<endl;

fi.open("usr.dat", ios::binary);

while(fi.read((char \*) &usr, sizeof(usr)))

{

cout << "Username : " <<usr.Username <<endl;

cout << "Email : " <<usr.Email <<endl;

cout << "Date Join : " <<usr.DateJoined<<endl;

}

fi.close();

cout <<"Input Email To Deleted the data:";

cin>>email;

cout << "==============================" <<endl;

fi.open("usr.dat", ios::binary);

fo.open("tmp.dat", ios::out|ios::binary);

int fd = 0;

while(fi.read((char\*) &usr, sizeof(usr)))

{

if(strcmp(usr.Email, email) == 0)

{

fd = 1;

}

else{

fo.write((char\*)&usr, sizeof(usr));

}

}

fi.close();

fo.close();

remove("usr.dat");

rename("tmp.dat", "usr.dat");

if (fd ==1)

{

cout<<"Email " << email << " Has Been Deleted!\n";

}

else{

cout<<"ID " << email << " Not Founded!\n";

}

}

void newdata(){

fi.open("usr.dat", ios::binary);

cout << "Username : " <<usr.Username <<endl;

cout << "Email : " <<usr.Email <<endl;

cout << "Date Join : " <<usr.DateJoined<<endl;

fi.close();

}

void listdata(){

cout << "=============USER=============" <<endl;

fi.open("usr.dat", ios::binary);

while(fi.read((char \*) &usr, sizeof(usr)))

{

cout << "Username : " <<usr.Username <<endl;

cout << "Email : " <<usr.Email <<endl;

cout << "Date Join : " <<usr.DateJoined<<endl;

}

fi.close();

cout <<"Input Email To View Info:";

cin>>email;

fstream f;

f.open("usr.dat", ios::in|ios::out );

while(f.read((char \*) &usr, sizeof(usr))){

if(strcmp(usr.Email, email) == 0) {

//view all detail

cout << "Username : " <<usr.Username <<endl;

cout << "Email : " <<usr.Email <<endl;

cout << "Date Join : " <<usr.DateJoined<<endl;

code=usr.id;

}

}

f.close();

code=code;

f.open("member.dat", ios::in|ios::out );

while(f.read((char \*) &mbr, sizeof(mbr))){

if(mbr.id, code == 0) {

//view all detail

cout << "Level : " <<mbr.Level<<endl;

}

}

f.close();

f.open("specialty.dat", ios::in|ios::out );

while(f.read((char \*) &tnr, sizeof(tnr))){

if(code, tnr.id == 0) {

//view all detail

cout << "Specialty : " <<tnr.Specialty<<endl;

}

}

f.close();

}

int main()

{

no=0;

do{

fi.open("usr.dat", ios::binary);

while(fi.read((char \*) &usr, sizeof(usr)))

{

no=usr.id;

}

fi.close();

no=no+1;

cout << "=============Menu=============" <<endl<<endl;

cout << "|A / 1| To Add a new user." <<endl;

cout << "|E / 2| To Edit an exiting user." <<endl;

cout << "|D / 3| To Deleted an exiting user." <<endl;

cout << "|V / 4| To View new user." <<endl;

cout << "|L / 5| To List All user." <<endl;

cout << "|M / 6| (Menu) To perform the tasks from 1 – 5." <<endl<<endl;

cout <<"Select : ";

cin>>choose\_menu;

if(choose\_menu == '1' or choose\_menu=='A' or choose\_menu=='a'){

create();

}

else if(choose\_menu == '2' or choose\_menu=='E' or choose\_menu=='e'){

}

else if(choose\_menu == '3' or choose\_menu=='D' or choose\_menu=='d'){

deletedata();

}

else if(choose\_menu == '4' or choose\_menu=='V' or choose\_menu=='v'){

newdata();

}

else if(choose\_menu == '5' or choose\_menu=='L' or choose\_menu=='l'){

listdata();

}

else if(choose\_menu == '6' or choose\_menu=='M' or choose\_menu=='m'){

create();

deletedata();

newdata();

listdata();

}

else{

cout<<"not menu selected"<<endl<<endl;

}

cout<<"Do you wish to Restart ? (y/n)??";

cin>>r;

system ("CLS");

}

while(r=='y' or r=='Y');

//end if

}

**SCREENSHOT:**







