



UNIVERSITY MANAGEMENT SYSTEM



SUBMITTED TO:
Engr. REHAN AHMED
SUBMITTED BY:
NAME: AROOJ SALEEM
ROLL # 2023-BSE-013

Fatima Jinnah Women University

Introduction:

The provided C++ code implements a comprehensive University Management System designed for managing various administrative and academic aspects of a university. The system allows for the management of different user roles, including students, professors, and staff members, each with their own specific functionalities. The **Person** class serves as a base class for storing common attributes like name and CNIC, while derived classes such as **Student**, **Professor**, and **Staff** extend it to include role-specific information and behaviors. Additionally, the system incorporates features for calculating and displaying student GPAs and CGPAs, setting and displaying salaries for professors and staff, managing fee structures for students, and maintaining a timetable. The ID class facilitates user authentication and registration, ensuring secure access to the system. Through a command-line interface, users can interact with the system to perform various tasks based on their roles, making it a versatile tool for university administration.

Purpose:

The purpose of this University Management System (UMS) code is to automate and streamline various administrative and academic processes within a university. The code is designed to manage and facilitate the following key functions:

1. User Authentication and Registration:

Provides functionality for new users to register by setting up a username and password. Allows existing users to log in using their credentials. Supports password updates and secure login processes.

1. Student Management:

Collects and stores student details such as name, CNIC, date of birth, roll number, semester, section, and department. Manages course information, including course titles, marks, and credit hours. Calculates GPA and CGPA based on the entered course details and marks. Displays student information, GPA, and fee structure based on the department and student type (regular or self-support).

2. Professor Management:

Collects and stores professor details such as name, CNIC, designation, qualification, and department. Sets and displays salary information based on the professor's designation. Manages and displays professor-specific details and timetable.

3. Staff Management:

Collects and stores staff details such as name, CNIC, designation, hire date, and department. Sets and displays salary information based on the staff member's designation. Manages and displays staff-specific details.

4. Department Management:

Manages department-specific functions, including setting timetables and calculating CGPA for students. Displays department information, timetables, and CGPA calculations.

5. Administrative Functions:

Enables the administration to manage and set student fees, professor salaries, and staff salaries. Allows setting and viewing timetables for various departments.

Overall, the code aims to provide a unified platform for managing university operations, reducing the need for manual data handling, and improving efficiency in handling student, professor, and staff information. It ensures that different roles within the university can access and manage relevant information securely and effectively.

Code:

```
#include "stdafx.h"
#include <iostream>
#include <fstream>
#include <string>
using namespace std;
class Person {
protected:
string name, cnic;
public:
Person() : name("\0"), cnic("\0") {}
Person(string n, string c) : name(n), cnic(c) {}
void get_data() {
cout << "\nEnter Name: ";
cin.ignore();
getline(cin, name);
cout << "\nEnter CNIC (format XXXXX-XXXXXXX-X): ";
getline(cin, cnic);
}
void display() {
cout << "\nYour Name: " << name << endl << "\nYour CNIC # " << cnic << endl;
}
};
class Admin { //abstract class
public:
virtual void set_salary()=0;
virtual void set_fee() =0;
};
class Student : public Person ,public Admin{
protected:
string roll_no, section, department, course_title[10], dob, grade[10];
int semester, no_of_courses;
double marks[10], gpa[10], credit_hours[10];
```

```
char c;
public:
int get_no_of_courses(){
return no_of_courses;
}
void get_data() {
Person::get_data();
cout << "\tEnter date of birth: ";
cin >> dob;
cout << "\tEnter roll number: ";
cin >> roll_no;
cout << "\tEnter semester: ";
cin >> semester;
cin.ignore();
cout << "\tEnter section: ";
getline(cin, section);
cout << "\tEnter department: ";
getline(cin, department);
cout << "\tDo you want to check your GPA :";
cin >> c;
cin.ignore();
if(c=='y' || c=='Y'){
cout << "\tEnter Number of Courses: ";
cin >> no_of_courses;
cin.ignore();
for (int i = 0; i < no_of_courses; i++) {
cout << "\tEnter the course title: ";
getline(cin, course_title[i]);
cout << "\tPlease enter your marks of this course: ";
cin >> marks[i];
cout << "\tEnter your Credit hours of this Course: ";
cin >> credit_hours[i];
cin.ignore();
}
cal_gpa();
} else
no_of_courses = 0;
}
double* get_gpa(){
return gpa;
}
double* get_credit_hours(){
return credit_hours;
}
void cal_gpa() {
if(no_of_courses >0){
for (int i = 0; i < no_of_courses; i++) {
if (marks[i] >= 85){
```

```
gpa[i] = 4.0;
grade[i] = "A";
else if (marks[i] >= 80){
    gpa[i] = 3.7;
    grade[i] = "A-";
}
else if (marks[i] >= 75){
    gpa[i] = 3.3;
    grade[i] = "B+";
}
else if (marks[i] >= 70){
    gpa[i] = 3.0;
    grade[i] = "B";
}
else if (marks[i] >= 65){
    gpa[i] = 2.7;
    grade[i] = "B-";
}
else if (marks[i] >= 60){
    gpa[i] = 2.3;
    grade[i] = "C+";
}
else if (marks[i] >= 58){
    gpa[i] = 2.0;
    grade[i] = "C";
}
else if (marks[i] >= 55){
    gpa[i] = 1.7;
    grade[i] = "C-";
}
else if (marks[i] >= 50){
    gpa[i] = 1.5;
    grade[i] = "D";
}
else{
    gpa[i] = 0.0;
    grade[i] = "F";
}
}
}
else
cout << "\t You have entered no of courses 0 so there is no GPA calculated " << endl;
}
void set_fee() {
cout << "\t If you are a Regular Student press 1, for Self support press 2: ";
cin >> c;
if (department == "Software engineering" || department == "SE") {
if (c == '1')
cout << "\t Your Total Fee is 50,000 Rs" << endl;
else if (c == '2')
cout << "\t Your Total Fee is 95,000 Rs" << endl;
} else if (department == "Computer Science" || department == "CS") {
if (c == '1')
cout << "\t Your Total Fee is 50,000 Rs" << endl;
```

```

else if (c == '2')
cout << "\tYour Total Fee is 85,000 Rs" << endl;
} else if (department == "Electronic engineering" || department == "EE") {
if (c == '1')
cout << "\tYour Total Fee is 50,000 Rs" << endl;
else if (c == '2')
cout << "\tYour Total Fee is 85,000 Rs" << endl;
} else if (department == "Business Administration" || department == "BA") {
if (c == '1')
cout << "\tYour Total Fee is 50,000 Rs" << endl;
else if (c == '2')
cout << "\tYour Total Fee is 85,000 Rs" << endl;
} else if (department == "Physics" || department == "Phy") {
if (c == '1')
cout << "\tYour Total Fee is 50,000 Rs" << endl;
else if (c == '2')
cout << "\tYour Total Fee is 85,000 Rs" << endl;
} else if (department == "Education" || department == "Edu") {
if (c == '1')
cout << "\tYour Total Fee is 50,000 Rs" << endl;
else if (c == '2')
cout << "\tYour Total Fee is 85,000 Rs" << endl;
} else
cout << "\tDepartment not recognized!" << endl;
}
void set_salary(){
//no implementation required in this class
}
void display() {
cout << "\tYou have entered the following information:" << endl;
Person::display();
cout << "\tDate of Birth: " << dob << endl << "\tRoll #: " << roll_no << endl << "\tSemester: " <<
semester << endl << "\tSection: " << section << endl << "\tDepartment: " << department << endl;
if (no_of_courses > 0) {
cout << "\tGPA for the courses:" << endl;
for (int i = 0; i < no_of_courses; i++) {
cout << "\tYou have " << gpa[i] << " GPA in the Course " << course_title[i] << endl << "\tYour Grade is
" << grade[i] << " in the Course " << course_title[i] << endl;
}
}
else
cout << "\tNo GPA Calculated " << endl;
}
};

class Professor : public Person, public Admin {
protected:

```

```
string designation, qualification, department;
double salary, medical, house_rent ;
public:
void get_data() {
Person::get_data();
cout << "\tEnter your designation: ";
getline(cin, designation);
cout << "\tEnter qualification: ";
getline(cin, qualification);
cout << "\tEnter Department: ";
getline(cin, department);
}
void set_salary() {
if (designation == "Permanent Instructor" || designation == "permanent instructor") {
salary = 200000;
} else if (designation == "Visiting Instructor" || designation == "visiting instructor") {
salary = 150000;
} else if (designation == "Contractual Instructor" || designation == "contractual instructor") {
salary = 170000;
} else
cout << "\tDesignation not Recognized! " << endl;
medical = 0.25 * salary;
house_rent = 0.1 * salary;
}
void set_fee(){
//no implementation required in this class
}
void display() {
cout << "\tYou have entered the following information:" << endl;
Person::display();
cout << "\tYour Designation : " << designation << "\n\tYour Qualification : " << qualification <<
"\n\tYour Department : " << department << endl;
cout << "\tYour Salary is " << (salary + medical + house_rent) << endl;
}
};

class Staff : public Admin, public Person {
protected:
string designation, department, hire_date;
double salary, house_rent, medical ;
public:
void get_data() {
Person::get_data();
cout << "\tEnter hire date: ";
getline(cin, hire_date);
cout << "\tEnter Department: ";
getline(cin, department);
```

```
cout << "\tEnter your designation: ";
getline(cin, designation);
}
void set_salary() {
if (designation == "DC" || designation == "dc") {
salary = 150000;
} else if (designation == "HOD" || designation == "hod") {
salary = 300000;
} else if (designation == "lab assistant" || designation == "Lab Assistant") {
salary = 100000;
} else if (designation == "security guard" || designation == "Security Guard") {
salary = 50000;
} else if (designation == "cleaner" || designation == "Cleaner") {
salary = 30000;
} else if (designation == "engineer" || designation == "Engineer") {
salary = 150000;
} else {
cout << "\tDesignation not recognized!" << endl;
return;
}
medical = 0.25 * salary;
house_rent = 0.1 * salary;
}
void set_fee() {
// No implementation required in this class
}
void display() {
cout << "\tYou have entered the following information:" << endl;
Person::display();
cout << "\tDesignation: " << designation << "\n\tDepartment: " << department << "\n\tHire Date: " <<
hire_date << endl;
cout << "\tYour Salary is " << (salary + medical + house_rent) << endl;
}
};

class Department {
protected:
string timetable[5][5];
double cgpa;
Student* S;
Professor* P;
public:
Department(Student* sobj, Professor* pobj){
this->S=sobj;
this->P=pobj;
}
void get_data() {
```

```
char c, x;
cout << "Are you a student (y/n)? ";
cin >> c;
if (c == 'y' || c == 'Y') {
S->get_data();
cout << "\tDo you want to check your CGPA (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
CGPA();
cout << "\tDo you want to check your Semester Fee (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
S->set_fee();
} else if (c == 'n' || c == 'N') {
// If not a student, check if the user is a professor
cout << "Are you a Professor (y/n)? ";
cin >> c;
if (c == 'y' || c == 'Y') {
P->get_data();
cout << "\tDo you want to check your Salary (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
P->set_salary();
} else {
cout << "Invalid option selected!" << endl;
}
} else {
cout << "Invalid option selected!" << endl;
}
}

void set_timetable() {
cout << "Enter the timetable (3 rows for days, 3 columns for time slots):" << endl;
cin.ignore();
for (int i = 0; i < 3; i++) {
cout << "Day " << i + 1 << ":" << endl;
for (int j = 0; j < 3; j++) {
cout << "\tSlot " << j + 1 << " (Enter time and details): ";
getline(cin, timetable[i][j]);
}
}
}

void CGPA() {
S->cal_gpa();
double creditPoints = 0, creditHour = 0;
if (S->get_no_of_courses() > 0) {
```

```
for (int i = 0; i < S->get_no_of_courses(); i++) {
    creditPoints = creditPoints + ( S->get_gpa()[i] * S->get_credit_hours()[i]);
    creditHour = creditHour + S->get_credit_hours()[i];
}
cgpa = creditPoints / creditHour;
}
else
cgpa = 0.0;
}
void display() {
char c;
cout << "Are you a Student (y/n)? ";
cin >> c;
if (c == 'y' || c == 'Y') {
S->display();
cout << "\tYour CGPA is: " << cgpa << endl;
cout << "Timetable: " << endl;
for (int i = 0; i < 3; i++) {
cout << "Day " << i + 1 << ":";
for (int j = 0; j < 3; j++) {
cout << " " << timetable[i][j];
}
cout << endl;
}
} else if (c == 'n' || c == 'N') {
cout << "Are you a Professor (y/n)? ";
cin >> c;
if (c == 'y' || c == 'Y') {
P->display();
cout << "Timetable: " << endl;
for (int i = 0; i < 3; i++) {
cout << "Day " << i + 1 << ":";
for (int j = 0; j < 3; j++) {
cout << " " << timetable[i][j];
}
cout << endl;
}
} else {
cout << "Invalid option selected!" << endl;
}
} else {
cout << "Invalid option selected!" << endl;
}
}
};
```

```
class User {  
private:  
string username[10], password[10], pass;  
char ch;  
public:  
User(){  
for(int i=0; i<10; i++){  
username[i] = "";  
password[i] = "";  
}  
}  
void set_data(){  
for(int i=0;i<10;i++){  
if(username[i]== ""){  
cout<<"Set your username: ";  
cin>>username[i];  
cout<<"Set your password: ";  
cin>>password[i];  
cout << "User registered successfully." << endl;  
storeData(username[i],password[i]);  
loadData();  
return;  
}  
}  
}  
void storeData( string uname,string pwd) {  
ofstream fout("userID.txt", ios::app);  
if (fout.is_open()) {  
fout << uname << " " << pwd << endl;  
fout.close();  
} else {  
cout << "Unable to open file for writing." << endl;  
}  
}  
void loadData() {  
ifstream fin("userID.txt");  
string uname, pwd;  
int index = 0;  
while (fin >> uname >> pwd) {  
if (index < 10) {  
username[index] = uname;  
password[index] = pwd;  
index++;  
}  
}
```

```
fin.close();
}
void updateFile() {
ofstream fout("userID.txt");
if (fout.is_open()) {
for (int i = 0; i < 10; i++) {
if (username[i] != "") {
fout << username[i] << " " << password[i] << endl;
}
}
fout.close();
} else {
cout << "Unable to open file for writing." << endl;
}
}
bool login(string uname, string pwd) {
loadData();
bool found = false;
for (int i = 0; i < 10; i++) {
if (uname == username[i]) {
found = true;
if (pwd == password[i]) {
cout << "Login successful" << endl;
return true;
}
else {
cout << "Invalid password. Do you want to change your password? (y/n): ";
cin >> ch;
if (ch == 'y') {
cout << "Enter new password: ";
cin >> pass;
password[i] = pass;
updateFile();
cout << "Your password is updated successfully." << endl;
return true;
}
else {
cout << "Thank you for the response." << endl;
return false;
}
}
}
}
if (!found) {
cout << "Username not found." << endl;
```

```
return false;
}
}
};
void afterlogin()
{
Admin *a0,*a1,*a2;
Student s;
char c, x;
cout << "Are you Student? (press 1), Professor? (press 2), Staff? (press 3) , From department? (press4): ";
cin >> c;
if (c == '1') {
cout << "-----Student Class-----" << endl;
a0=&s;
s.get_data();
cout << "\tDo you want to check your Semester Fee (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
s.set_fee();
cout << "\tDo you want to check your Result (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
s.cal_gpa();
s.display();
}
else if (c == '2') {
cout << "-----Professor Class-----" << endl;
Professor p;
a1=&p;
p.get_data();
cout << "\tDo you want to check your Salary (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
p.set_salary();
p.display();
}
else if (c == '3') {
cout << "-----Staff Class-----" << endl;
Staff st;
a2=&st;
st.get_data();
cout << "\tDo you want to check your Salary (y/n)? ";
cin >> x;
if(x=='y' || x=='Y')
st.set_salary();
```

```
st.display();
}
else if(c=='4'){
cout<<"-----Department Class-----"<<endl;
Student s;
Professor p;
Department d(&s,&p);
d.set_timetable();
d.get_data();
system("cls");
d.display();
}
else {
cout << "\tInvalid option selected!" << endl;
}
return;
}
int main() {
User person;
int choice;
string username, password;
cout << "\t\t\t-----Welcome to FATIMA JINNAH WOMEN UNIVERSITY----- " << endl;
cout << "New User(press 1) or have an account(press 2)?";
cin>>choice;
if(choice==1){
person.set_data();
cout << "\nnow login to your account"<<endl;
cout << "Enter Username: ";
cin>>username;
cout << "Enter Password: ";
cin>>password;
if(person.login(username,password)==true)
afterlogin();
}
else if(choice==2){
cout << "Username: ";
cin>>username;
cout << "password: ";
cin>>password;
if(person.login(username,password)==true)
afterlogin();
}
system("pause");
return 0;
}
```

Output:

```
C:\Users\sp\onedrive\documents\visual studio 2010\Projects\University Management system\Debug\University Management system.exe
-----Welcome to FATIMA JINNAH WOMEN UNIVERSITY-----
New User(press 1) or have an account(press 2)?2
Username: arooj
password: arooj123
Login successful
Are you Student? (press 1), Professor? (press 2), Staff? (press 3) , From department? (press4): 1
-----Student Class-----
    Enter Name: arooj saleem
    Enter CNIC (format XXXXX-XXXXXXX-X): 12345-5678900-0
    Enter date of birth: 21/2/2005
    Enter roll number: 2023-bse-013
    Enter semester: 2
    Enter section: A
    Enter department: SE
    Do you want to check your GPA :y
    Enter Number of Courses: 2
    Enter the course title: oop
    Please enter your marks of this course: 90
    Enter your Credit hours of this Course: 2
    Enter the course title: cald
    Please enter your marks of this course: 84
    Enter your Credit hours of this Course: 3
    Do you want to check your Semester Fee (y/n)? y
    If you are a Regular Student press 1, for Self support press 2: 1
    Your Total Fee is 50,000 Rs
    Do you want to check your Result (y/n)? y
    You have entered the following information:
    Your Name: arooj saleem
    Your CNIC # 12345-5678900-0
    Date of Birth: 21/2/2005
    Roll #: 2023-bse-013
    Semester: 2
    Section: A
    Department: SE
    GPA for the courses:
    You have 4 GPA in the Course oop
    Your Grade is A in the Course oop
    You have 3.7 GPA in the Course cald
    Your Grade is A- in the Course cald
Press any key to continue . . .
```

Fatima Jinnah Women University

```
C:\Users\sp\onedrive\documents\visual studio 2010\Projects\University Management system\Debug\University Management system.exe
-----Welcome to FATIMA JINNAH WOMEN UNIVERSITY-----
New User(press 1) or have an account(press 2)?2
Username: arooj
password: arooj123
Login successful
Are you Student? (press 1), Professor? (press 2), Staff? (press 3) , From department? (press4): 2
-----Professor Class-----
    Enter Name: Sadia Ali
    Enter CNIC (format XXXXX-XXXXXXX-X): 12345-5678990-0
    Enter your designation: permanent instructor
    Enter qualification: P.H.D
    Enter Department: SE
    Do you want to check your Salary (y/n)? y
    You have entered the following information:
    Your Name: Sadia Ali
    Your CNIC # 12345-5678990-0
    Your Designation : permanent instructor
    Your Qualification : P.H.D
    Your Department : SE
    Your Salary is 270000
Press any key to continue . . .
```

```
C:\Users\sp\onedrive\documents\visual studio 2010\Projects\University Management system\Debug\University Management system.exe
-----Welcome to FATIMA JINNAH WOMEN UNIVERSITY-----
New User(press 1) or have an account(press 2)?2
Username: arooj
password: arooj
Invalid password. Do you want to change your password? (y/n): y
Enter new password: 123
Your password is updated successfully.
Are you Student? (press 1), Professor? (press 2), Staff? (press 3) , From department? (press4): 3
-----Staff Class-----
    Enter Name: saliha suleman
    Enter CNIC (format XXXXX-XXXXXXX-X): 12345-5678900-0
    Enter hire date: 21/2/2023
    Enter Department: SE
    Enter your designation: HOD
    Do you want to check your Salary (y/n)? y
    You have entered the following information:
    Your Name: saliha suleman
    Your CNIC # 12345-5678900-0
    Designation: HOD
    Department: SE
    Hire Date: 21/2/2023
    Your Salary is 405000
Press any key to continue . . .
```

Fatima Jinnah Women University

```
C:\Users\sp\onedrive\documents\visual studio 2010\Projects\University Management system\Debug\University Management system.exe
-----Welcome to FATIMA JINNAH WOMEN UNIVERSITY-----
New User(press 1) or have an account(press 2)?2
Username: arooj
password: 123
Login successful
Are you Student? (press 1), Professor? (press 2), Staff? (press 3) , From department? (press4): 4
-----Department Class-----
Enter the timetable (3 rows for days, 3 columns for time slots):
Day 1:
    Slot 1 (Enter time and details): oop(9:30-11:00)
    Slot 2 (Enter time and details): oop lab(11:00-2:00)
    Slot 3 (Enter time and details): se(2:30-4:00)
Day 2:
    Slot 1 (Enter time and details): cald(9:30-11:00)
    Slot 2 (Enter time and details): cald lab(11:00-2:00)
    Slot 3 (Enter time and details): disc(2:30-4:00)
Day 3:
    Slot 1 (Enter time and details): calc(9:30-11:00)
    Slot 2 (Enter time and details): calc lab(11:00-2:00)
    Slot 3 (Enter time and details): se(2:30-4:00)
Are you a student (y/n)? n
Are you a Professor (y/n)? y
    Enter Name: arooj
    Enter CNIC (format XXXXX-XXXXXXX-X): 12345-5678900-0
    Enter your designation: permanent instructor
    Enter qualification: mphil
    Enter Department: SE
    Do you want to check your Salary (y/n)? y
Are you a Student (y/n)? n
Are you a Professor (y/n)? y
    You have entered the following information:
    Your Name: arooj
    Your CNIC # 12345-5678900-0
    Your Designation : permanent instructor
    Your Qualification : mphil
    Your Department : SE
    Your Salary is 270000
Timetable:
Day 1: oop(9:30-11:00) oop lab(11:00-2:00) se(2:30-4:00)
Day 2: cald(9:30-11:00) cald lab(11:00-2:00) disc(2:30-4:00)
Day 3: calc(9:30-11:00) calc lab(11:00-2:00) se(2:30-4:00)
Press any key to continue . . .
```
