QUESTION 1:

MAIN:

#include<iostream>

#include<string>

#include<cstring>

#include"Header.h"

using namespace std;

int main() {

Person p;

p.setName("saad");

p.setAge(10);

p.setGender('M');

p.print();

Faculty f;

f.setFid(100);

f.setCourse("OOP");

f.print();

Student s;

s.setBatch("21st");

s.setSid(123);

s.print();

Admin a;

a.setDepartment("CS");

a.print();

system("pause");

}

HEADER:

#pragma once

#include<iostream>

#include<string>

#include<cstring>

using namespace std;

class Person {

private:

string name;

int age;

char gender;

public:

Person() {

name = "";

age = 0;

gender = 'a';

}

void setName(string nname) {

name = nname;

}

void setAge(int aage) {

age = aage;

}

void setGender(char ggender) {

gender = ggender;

}

string getName() {

return name;

}

int getAge() {

return age;

}

char gerGender() {

return gender;

}

void print() {

cout << "Name: " << name << endl;

cout << "Age : " << age << endl;

cout << "Gender : " << gender << endl;

}

};

class Faculty : public Person {

private:

string course;

int fid;

public:

void setCourse(string ccourse) {

course = ccourse;

}

void setFid(int ffid) {

fid = ffid;

}

string getCourse() {

return course;

}

int getFid() {

return fid;

}

void print() {

cout << "Fid : " << fid << endl;

cout << "Course : " << course << endl;

}

};

class Student {

private:

int sid;

string batch;

public:

void setBatch(string bbatch) {

batch = bbatch;

}

void setSid(int ssid) {

sid = ssid;

}

string getBatch() {

return batch;

}

int getSid() {

return sid;

}

void print() {

cout << "Sid : " << sid << endl;

cout << "Batch : " << batch << endl;

}

};

class Admin :public Faculty, public Student {

private:

string department;

public:

void setDepartment(string ddepartment) {

department = ddepartment;

}

string getDepartment() {

return department;

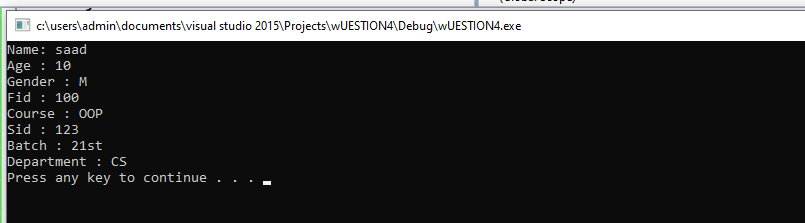
}

void print() {

cout << "Department : " << department << endl;

}

};



QUESTION 2:

#include<iostream>

using namespace std;

int isPalindrome(char str[], int first, int last) {

if (first < last + 1) {

first++;

last--;

return isPalindrome(str, first, last);

}

if (first == last) {

cout << "is palindrome";

return 1;

}

if (str[first] != str[last]) {

cout << "is palindrome" << endl;

return 0;

}

else {

cout << "is not Palindrome" << endl;

return true;

}

return 1;

}

int main() {

char Str[] = "sad";

int result;

int length = strlen(Str);

if (length == 0) {

result = 1;

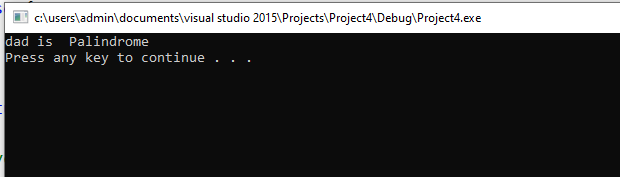
}

Cout<< “dad”;

isPalindrome("dad", 2, 5);

system("pause");

}



QUESTION 3:

#include<iostream>

using namespace std;

void input(int \*\*p, int row, int col) {

}

void display(int \*\*p, int row, int col) {

}

int\*\* multiply(int \*\*p, int row, int col, int \*\*q, int row2, int col2) {

}

int main() {

int rows1, coloumns1;

int rows2, coloumns2;

cout << "enter number of rows of matrix A : ";

cin >> rows1;

cout << "enter number of coloumns of matrix A : ";

cin >> coloumns1;

cout << "enter number of rows of matrix B : ";

cin >> rows2;

cout << "enter number of coloumns of matrix B : ";

cin >> coloumns2;

while (coloumns1 != rows2) {

cout << "error columns of matrix A must be equal to rows of matrix B";

cin >> coloumns1;

cin >> rows2;

}

}

Question 4:

MAIN:

#include<iostream>

#include<string>

#include<cstring>

#include"Header.h"

using namespace std;

int main() {

PersonBio p;

p.setName("saad");

p.setProfession\_name("CEO");

p.setAge(18);

p.print();

p.~PersonBio();

Address a;

cout << endl;

a.sethousenumber(123);

a.setstreetnumber(456);

a.setapartmentnumber(789);

a.setpostalcode(92);

a.setcityname("Lahore");

a.setstatename("Punjab");

a.print();

a.inputfunction();

a.print();

a.~Address();

Address a1;

a1.inputfunction();

a1.print();

a.~Address();

system("pause");

}

HEADER:

#pragma once

#include<iostream>

#include<string>

#include<cstring>

using namespace std;

class PersonBio {

private:

string name;

string profession\_name;

int age;

public:

PersonBio() {

name = "";

profession\_name = "";

age = 0;

}

void setName(string nname) {

name = nname;

}

void setProfession\_name(string pprofession\_name) {

profession\_name = pprofession\_name;

}

void setAge(int aage) {

age = aage;

}

string getName() {

return name;

}

string getProfession\_name() {

return profession\_name;

}

int getAge() {

return age;

}

void print() {

cout << "Name : " << name << endl;

cout << "Profession Name : " << profession\_name << endl;

cout << "Age : " << age << endl;

}

~PersonBio() {

cout << "i am destructor of personbio class and objects are finished" << endl;

}

};

class Address {

private:

int housenumber, streetnumber, apartmentnumber, postalcode;

string cityname, statename;

public:

Address() {

housenumber = 0;

streetnumber = 0;

apartmentnumber = 0;

postalcode = 0;

cityname = "";

statename = "";

cout << "I am Default Constructor of Address class";

}

Address(int a, int b, int c, int d, string e, string f) {

housenumber = a;

streetnumber = b;

apartmentnumber = c;

postalcode = d;

cityname = e;

statename = f;

cout << "I am Parameterized Constructor";

}

void sethousenumber(int g) {

housenumber = g;

}

void setstreetnumber(int h) {

streetnumber = h;

}

void setapartmentnumber(int i) {

apartmentnumber = i;

}

void setpostalcode(int j) {

postalcode = j;

}

void setcityname(string k) {

cityname = k;

}

void setstatename(string l) {

statename = l;

}

int gethousenumber() {

return housenumber;

}

int getstreetnumber() {

return streetnumber;

}

int getapartmentnumber() {

return apartmentnumber;

}

int getpostalcode() {

return postalcode;

}

string getcityname() {

return cityname;

}

string getstatename() {

return statename;

}

void inputfunction() {

cout << endl;

cout << "enter house number: ";

cin >> housenumber;

cout << "enter streetnumber: ";

cin >> streetnumber;

cout << "enter apartmentnumber: ";

cin >> apartmentnumber;

cout << "enter postalcode: ";

cin >> postalcode;

cout << "enter cityname: ";

cin >> cityname;

cout << "enter statename: ";

cin >> statename;

}

void print() {

cout << "house number : " << housenumber << " street number : " << streetnumber << " apartment number : " << apartmentnumber << endl;

cout << "city name : " << cityname << " state name: " << statename << " postal code: " << postalcode << endl;

}

~Address() {

cout << "i am destructor of Address class and objects are finished" << endl;

}

};

