HOMEWORK: OOP 3

NAME: Hoàng Trung Hiếu

CLASS: 20Se3

Bài 1:

Mã nguồn

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Exam

{

class Student

{

private int SID;

private string TenSV;

private string Khoa;

private float DiemTB;

public Student() //Constructor mac dinh

{

SID = 1;

TenSV = "Nguyen Van Nam";

Khoa = "CNTT";

DiemTB = 7;

}

public Student(Student stu) //Constructor sao chep

{

SID = stu.SID;

TenSV = stu.TenSV;

Khoa = stu.Khoa;

DiemTB = stu.DiemTB;

}

//Constructor tham so

public Student(int id, string ten, string kh, float dtb)

{

SID = id;

TenSV = ten;

Khoa = kh;

DiemTB = dtb;

}

//Cac Property cho tung thuoc tinh cua lop

public int StudentID //Property dai dien cho thuoc tinh SID

{

get { return SID; } //Lay du lieu

set { SID = value; } //Gan du lieu

}

public String Name

{

get { return TenSV; }

set { TenSV = value; }

}

public String Faculty

{

get { return Khoa; }

set { Khoa = value; }

}

public float Mark

{

get { return DiemTB; }

set { DiemTB = value; }

}

//Phuong thuc hien thi du lieu

public void Show()

{

Console.WriteLine("MSSV:{0}", this.SID);

Console.WriteLine("Ten SV:{0}", this.TenSV);

Console.WriteLine("Khoa:{0}", this.Khoa);

Console.WriteLine("Diem TB:{0}", this.DiemTB);

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Exam

{

class Tester

{

public static void Main()

{

Student[] DSSV;

int n;

Console.Write("Nhap so luong SV:");

n = int.Parse(Console.ReadLine());

DSSV = new Student[n]; //tao mang n phan tu

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

for (int i = 0; i < n; i++) //Lap n lan nhap thong tin sv

{

DSSV[i] = new Student();

Console.Write("Nhap MaSV {0}:", i + 1);

DSSV[i].StudentID = int.Parse(Console.ReadLine());

Console.Write("Ho ten SV:");

DSSV[i].Name = Console.ReadLine();

Console.Write("Nhap khoa:");

DSSV[i].Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

DSSV[i].Mark = float.Parse(Console.ReadLine());

}

//Xuat DS Sinh vien

Console.WriteLine("\n ====XUAT DS SINH VIEN====");

foreach (Student sv in DSSV)

sv.Show();

Console.ReadLine();

}

}

}

Biên dịch và chạy chương trình:

A screenshot of a computer

Description automatically generated

Debug

A computer screen shot of a program

Description automatically generated

A screenshot of a computer

Description automatically generated

Cửa sổ class view

A screenshot of a computer

Description automatically generated

Class view diagram

A computer screen shot of a computer

Description automatically generated

Bài 2:

Mã nguồn

using System;

namespace Exam2

{

class Student

{

private int SID;

private string TenSV;

private string Khoa;

private float DiemTB;

public Student()

{

SID = 1;

TenSV = "Nguyen Van A";

Khoa = "CNTT";

DiemTB = 7;

}

public Student(Student stu)

{

SID = stu.GetStudentID();

TenSV = stu.GetName();

Khoa = stu.GetFaculty();

DiemTB = stu.GetMark();

}

public Student(int id, string ten, string kh, float dtb)

{

SID = id;

TenSV = ten;

Khoa = kh;

DiemTB = dtb;

}

public int GetStudentID()

{

return SID;

}

public void SetStudentID(int id)

{

SID = id;

}

public string GetName()

{

return TenSV;

}

public void SetName(string ten)

{

TenSV = ten;

}

public string GetFaculty()

{

return Khoa;

}

public void SetFaculty(string kh)

{

Khoa = kh;

}

public float GetMark()

{

return DiemTB;

}

public void SetMark(float dtb)

{

DiemTB = dtb;

}

public void Show()

{

Console.WriteLine("MSSV:{0}", this.GetStudentID());

Console.WriteLine("Ten SV:{0}", this.GetName());

Console.WriteLine("Khoa:{0}", this.GetFaculty());

Console.WriteLine("Diem TB:{0}", this.GetMark());

}

}

}

using Exam2;

using System;

namespace Exam2

{

class Tester

{

static void Nhap1SV(Student sv)

{

Console.Write("Nhap MaSV: ");

sv.SetStudentID(int.Parse(Console.ReadLine()));

Console.Write("Ho ten SV: ");

sv.SetName(Console.ReadLine());

Console.Write("Nhap khoa: ");

sv.SetFaculty(Console.ReadLine());

Console.Write("Nhap Diem TB: ");

sv.SetMark(float.Parse(Console.ReadLine()));

}

static void NhapDS(Student[] DSSV)

{

int n = DSSV.Length;

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

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

{

DSSV[i] = new Student();

Console.Write("Nhap thong tin cho sinh vien {0}:\n", i + 1);

Nhap1SV(DSSV[i]);

}

}

static void XuatDS(Student[] DSSV)

{

Console.WriteLine("\n ====XUAT DS SINH VIEN====");

foreach (Student sv in DSSV)

sv.Show();

}

static void Main()

{

int n;

Console.Write("Nhap so luong SV: ");

n = int.Parse(Console.ReadLine());

Student[] DSSV = new Student[n];

NhapDS(DSSV);

XuatDS(DSSV);

Console.ReadLine();

}

}

}

A computer screen shot of a black screen

Description automatically generated

Bài 3

Mã nguồn

using System;

using System.Collections.Generic;

namespace Ex3

{

class People

{

private string Ten;

private string Khoa;

public People()

{

Ten = "Nguyen Van A";

Khoa = "CNTT";

}

public People(string ten, string kh)

{

Ten = ten;

Khoa = kh;

}

public string GetName()

{

return Ten;

}

public void SetName(string ten)

{

Ten = ten;

}

public string GetFaculty()

{

return Khoa;

}

public void SetFaculty(string kh)

{

Khoa = kh;

}

}

class Student : People

{

private int SID;

private float DiemTB;

public Student() : base()

{

SID = 1;

DiemTB = 7;

}

public Student(int id, string ten, string kh, float dtb) : base(ten, kh)

{

SID = id;

DiemTB = dtb;

}

public int GetStudentID()

{

return SID;

}

public void SetStudentID(int id)

{

SID = id;

}

public float GetMark()

{

return DiemTB;

}

public void SetMark(float dtb)

{

DiemTB = dtb;

}

public void Show()

{

Console.WriteLine("MSSV: {0}", this.GetStudentID());

Console.WriteLine("Ten SV: {0}", this.GetName());

Console.WriteLine("Khoa: {0}", this.GetFaculty());

Console.WriteLine("Diem TB: {0}", this.GetMark());

}

}

class Tester

{

static void Nhap1SV(Student sv)

{

Console.Write("Nhap MaSV: ");

sv.SetStudentID(int.Parse(Console.ReadLine()));

Console.Write("Ho ten SV: ");

sv.SetName(Console.ReadLine());

Console.Write("Nhap khoa: ");

sv.SetFaculty(Console.ReadLine());

Console.Write("Nhap Diem TB: ");

sv.SetMark(float.Parse(Console.ReadLine()));

}

static void NhapDS(List<Student> DSSV)

{

Console.Write("Nhap so luong SV: ");

int n = int.Parse(Console.ReadLine());

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

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

{

Student sv = new Student();

Console.Write("Nhap thong tin cho sinh vien {0}:\n", i + 1);

Nhap1SV(sv);

DSSV.Add(sv);

}

}

static void XuatDS(List<Student> DSSV)

{

Console.WriteLine("\n ====XUAT DS SINH VIEN====");

foreach (Student sv in DSSV)

sv.Show();

}

static void Main()

{

List<Student> DSSV = new List<Student>();

NhapDS(DSSV);

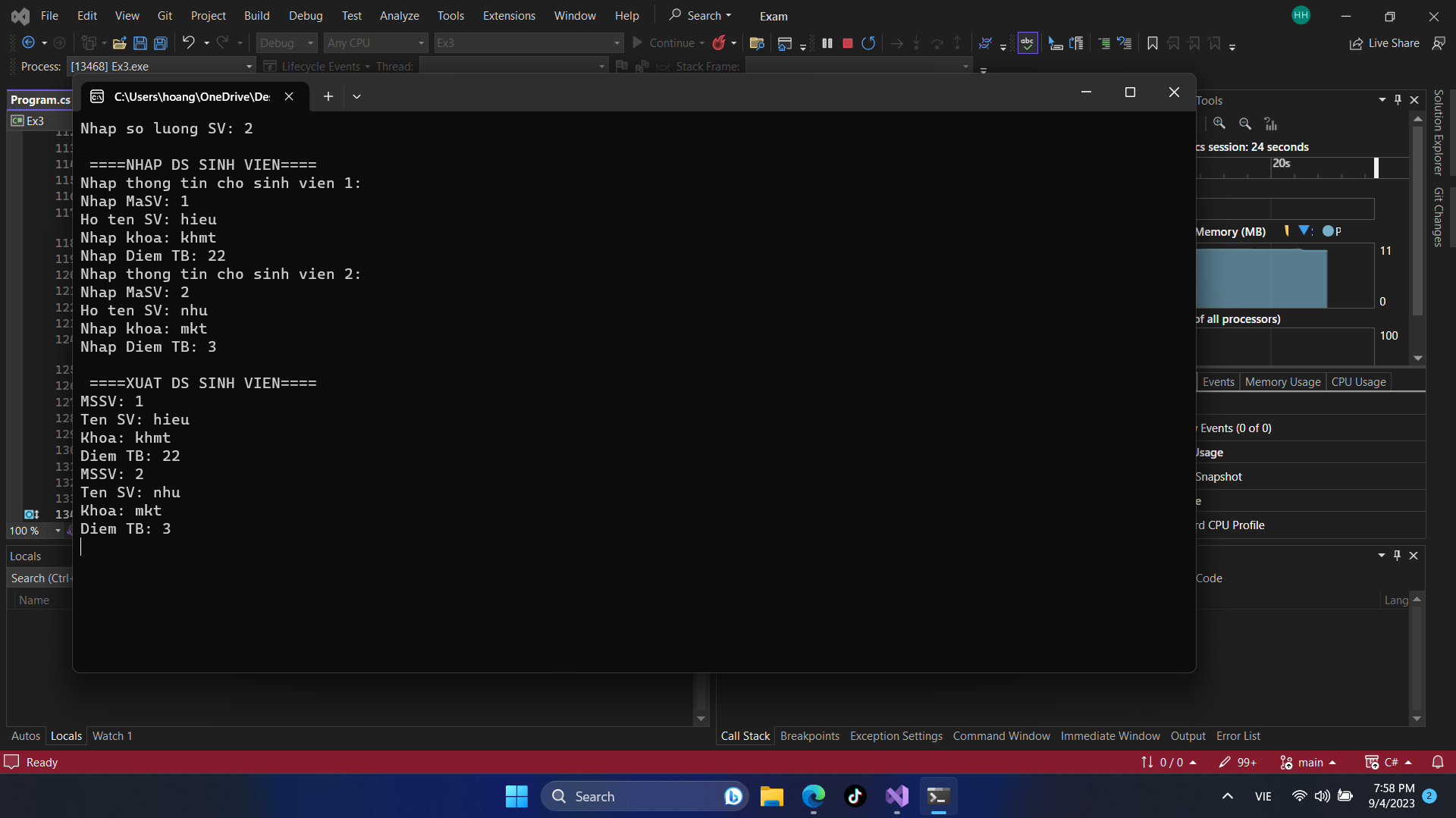
XuatDS(DSSV);

Console.ReadLine(); // Kết thúc chương trình sau khi xuất dữ liệu

}

}

}



Bài 4

Mã nguồn

using System;

using System.Collections.Generic;

class Person

{

public string FullName { get; set; }

public int BirthYear { get; set; }

public string Degree { get; set; }

public string Position { get; set; }

}

class Scientist : Person

{

public int PublishedPapers { get; set; }

public int WorkDaysPerMonth { get; set; }

public int SalaryGrade { get; set; }

public int CalculateSalary()

{

return WorkDaysPerMonth \* SalaryGrade;

}

}

class Manager : Person

{

public int WorkDaysPerMonth { get; set; }

public int SalaryGrade { get; set; }

public int CalculateSalary()

{

return WorkDaysPerMonth \* SalaryGrade;

}

}

class LabEmployee : Person

{

public int MonthlySalary { get; set; }

}

class Program

{

static void Main()

{

List<Person> instituteStaff = new List<Person>();

// Nhập thông tin các đối tượng

Console.WriteLine("Nhập thông tin về nhân viên viện khoa học:");

Scientist scientist1 = new Scientist

{

FullName = "Nguyen Van A",

BirthYear = 1980,

Degree = "Tiến sĩ",

Position = "Giáo sư",

PublishedPapers = 50,

WorkDaysPerMonth = 20,

SalaryGrade = 1000

};

instituteStaff.Add(scientist1);

Console.WriteLine("Nhập thông tin về nhân viên quản lý:");

Manager manager1 = new Manager

{

FullName = "Tran Thi B",

BirthYear = 1975,

Degree = "Thạc sĩ",

Position = "Trưởng phòng",

WorkDaysPerMonth = 22,

SalaryGrade = 800

};

instituteStaff.Add(manager1);

Console.WriteLine("Nhập thông tin về nhân viên phòng thí nghiệm:");

LabEmployee labEmployee1 = new LabEmployee

{

FullName = "Hoang Van C",

BirthYear = 1990,

Degree = "Cử nhân",

MonthlySalary = 600

};

instituteStaff.Add(labEmployee1);

// Xuất thông tin và tổng lương cho từng loại đ

Console.WriteLine("\nDanh sách nhân viên Viện Khoa học:");

foreach (var person in instituteStaff)

{

Console.WriteLine($"Họ tên: {person.FullName}");

Console.WriteLine($"Năm sinh: {person.BirthYear}");

Console.WriteLine($"Bằng cấp: {person.Degree}");

Console.WriteLine($"Chức vụ: {person.Position}");

Console.WriteLine();

if (person is Scientist scientist)

{

Console.WriteLine($"Số bài báo đã công bố: {scientist.PublishedPapers}");

Console.WriteLine($"Số ngày công trong tháng: {scientist.WorkDaysPerMonth}");

Console.WriteLine($"Tổng lương: {scientist.CalculateSalary()} VNĐ");

}

else if (person is Manager manager)

{

Console.WriteLine($"Số ngày công trong tháng: {manager.WorkDaysPerMonth}");

Console.WriteLine($"Tổng lương: {manager.CalculateSalary()} VNĐ");

}

else if (person is LabEmployee labEmployee)

{

Console.WriteLine($"Lương trong tháng: {labEmployee.MonthlySalary} VNĐ");

}

Console.WriteLine("-----------------------------------------");

}

// Tính tổng lương đã chi trả cho từng loại đối tượng

int totalScientistSalary = 0;

int totalManagerSalary = 0;

int totalLabEmployeeSalary = 0;

foreach (var person in instituteStaff)

{

if (person is Scientist scientist)

{

totalScientistSalary += scientist.CalculateSalary();

}

else if (person is Manager manager)

{

totalManagerSalary += manager.CalculateSalary();

}

else if (person is LabEmployee labEmployee)

{

totalLabEmployeeSalary += labEmployee.MonthlySalary;

}

}

Console.WriteLine($"Tổng lương của nhà khoa học: {totalScientistSalary} VNĐ");

Console.WriteLine($"Tổng lương của nhà quản lý: {totalManagerSalary} VNĐ");

Console.WriteLine($"Tổng lương của nhân viên phòng thí nghiệm: {totalLabEmployeeSalary} VNĐ");

}

}

