ASSIGNMENT 1 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Computing**		
Unit number and title	Unit 1: Programming		
Submission date		**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Bùi Hương Linh	**Student ID**	GBH200662
Class	GCH1002	**Assessor name**	Lecturer Mạnh

Student declaration

I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.

Student's signature	

Grading grid

P2	P3	P4	P5	M2	M3	M4	D2	D3	D4

☼Summative Feedback:		☼Resubmission Feedback:
Grade:	**Assessor Signature:**	**Date:**
Lecturer Signature:		

Contents

Task 1: Introduction to your program

I. Introduce the Overview

- Create a student management software program:
  The software includes personal information and scores for three subjects: math, physics, and chemistry. The program must sort and display the list of students on the screen.

II. List out application's requirements

- Fill in student information
- Math scores, physics grades, chemistry grades
- The app can add students, update, delete, and sort.
- The application is built on C sharp computer language
- It is win form application.

Task 2: Explain programming paradigms(P2)

I. Explain what is Procedural Programming with source code and illustrations

- Procedural programming (POP) is where the major focus on performing tasks in a sequential order. It divides a large program into small functional blocks or functions for ease of programming and testing easier.(Learn Computer Science,2021)

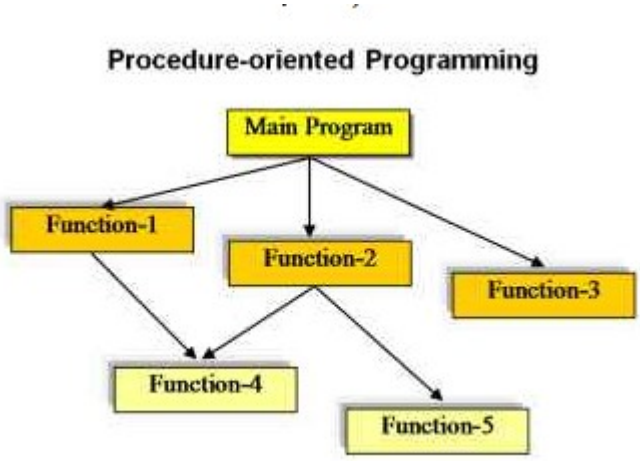


Image 1: Procedure Programming(source: Internet)

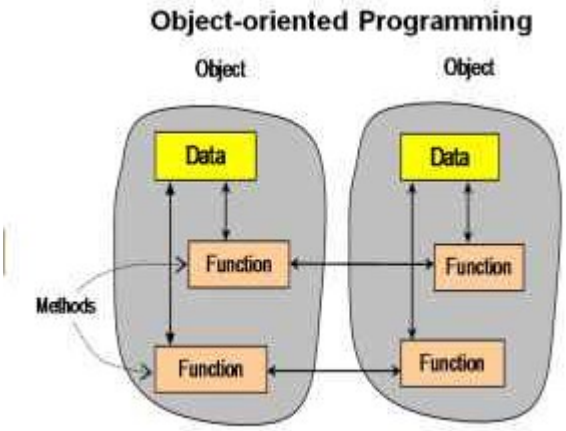
- Characteristies:
  - Focus on the work to be done (algorithms).
    + Helps beginners can improve their mindset about solving problems.
  - Large program is divided into subroutines, each of which can be called one or more times in any order.

+It makes it easier for programmers to address problems since faults in each sub-program may be readily fixed.

- Most functions use common data.
- Data in the system is moved from one function to another.
  + Programmers can manage data easily.
- Uses immutable data.(Leonila Cordrey,2021)

II. Explain what is Object-Oriented Programming with source code and illustrations

- Object-oriented programming (OOP) is the basic and most popular programming paradigm used by most developers. Object-oriented programming is a programming method based on the concept of classes and objects. It focuses on manipulating objects rather than the logic for manipulating them, making the code manageable, reusable, and maintainable.
- This programming approach was developed to reduce some of the drawbacks encountered in the Procedure Oriented Programming Approach.
  (Erin Doherty, 2021)



Imge 2: Object-oriented Programming(source: Internet)

- Characteristies: (Erin Doherty, 2021)

  - Encapsulation: This helps increase the security of the object and avoid the situation of data being unintentionally damaged.

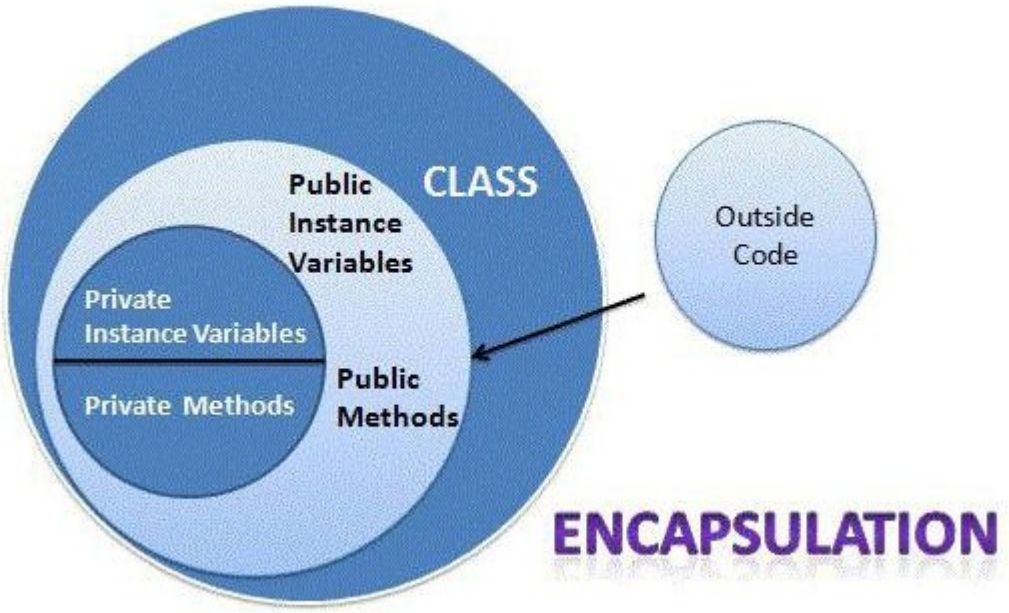


Image 3: Encapsulation (source: Internet)

- Polymorphism: Different objects can perform the same function in different ways.

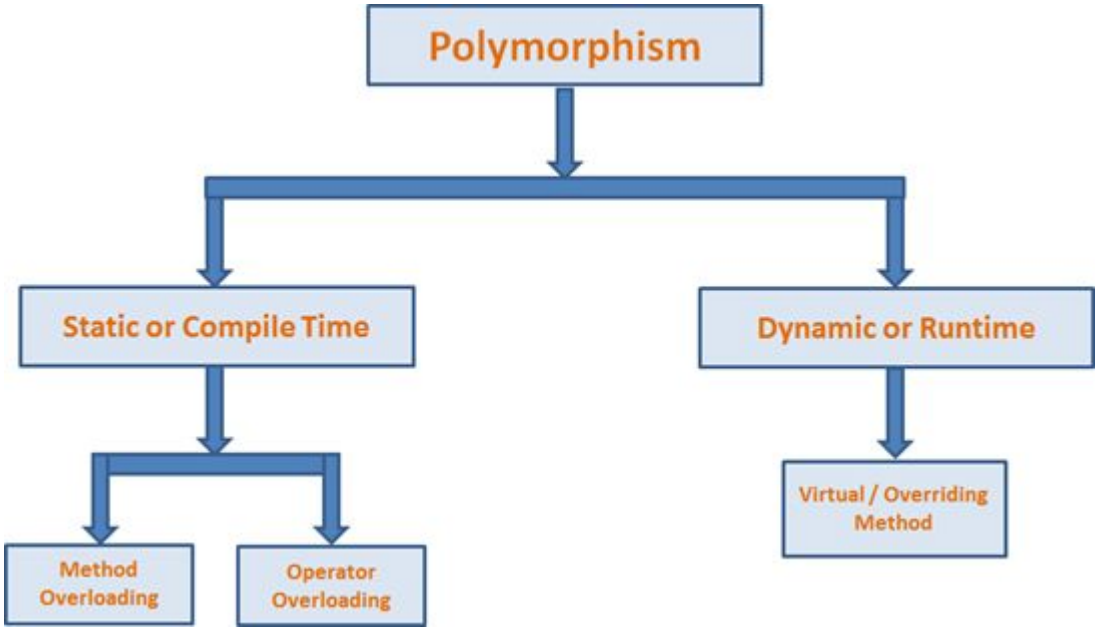


Image 4: Polymorphism (source: Internet)

- Inheritance: This is how a class can inherit properties and methods from another class and use them as its own.

```
class Animal
{
    protected double Weight;
    protected double Height;
    protected static int Legs;

    public void Info()
    {
        Console.WriteLine(" Weight: " + Weight + " Height: " + Height + " Legs: " + Legs);
    }
}
class Cat : Animal
{
    public Cat()
    {
        Weight = 500;
        Height = 20;
        Legs = 2;
    }
}
class Demo
{
    static void Main(string[] args)
    {
        Cat BlackCat = new Cat();

        /* Lớp Cat kế thừa phương thức Info từ lớp Animal nên đối tượng thuộc lớp Cat có thể gọi phương thức Info() */

        BlackCat.Info();
    }
}
```

Image 5: Inheritance

III. Explain what is Event-Driven Programming with source code and illustrations

- Event-driven programming is an important concept in application development and other programming styles, and it generates event handlers and other resources.(Copyright Techopedia, 2021)

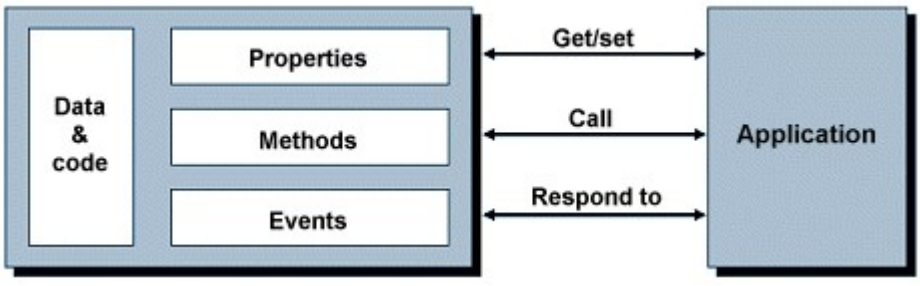


Image 6: Event-Driven Programming(source: Internet)

IV. Conclude which paradigms will be used to develop the application with explanation

- Object-oriented programming will be used to develop a student management application that I am about to do. Because we will use objects, classes, properties, storage, and computation methods to design and develop software.

Task 3: IDE features(P3-P4-P5)

I. Introduce what is IDE

- The IDE enables programmers to unify the various aspects of writing a computer program. It is also a software application that gathers all of the tools required for a software development project into a single location.(Aaron Walker, 2021)

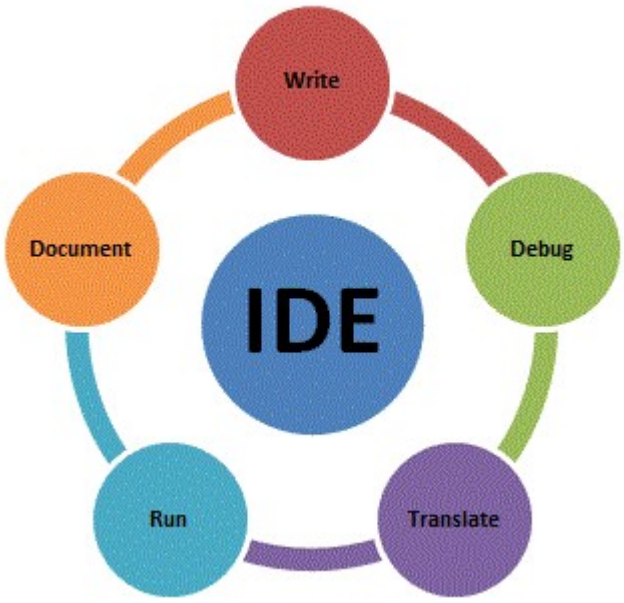


Image 7: IDE(source: Internet)

II. Introduce features of IDE with illustrations (Aaron Walker, 2021)

- Text editor: Virtually every IDE will have a text editor designed to write and manipulate source code. Some tools may have visual components to drag and drop front-end components, but most have a simple interface highlighting language-specific syntax.
- Debugger: Debugging tools assist users in identifying and remedying errors within source code. They often simulate real-world scenarios to test functionality and performance. Programmers and software engineers can usually test the various code segments and identify errors before the application is released.

- Compiler: Compilers are components that translate programming language into a form machines can process, such as binary code. The machine code is analyzed to ensure its accuracy. The compiler then parses and optimizes the code to optimize performance.
- Code completion: Code complete features assist programmers by intelligently identifying and inserting common code components. These features save developers time writing code and reduce the likelihood of typos and bugs.
- Programming language support: IDEs are typically specific to a single programming language, though several also offer multi-language support. As such, the first step is to figure out which languages you will be coding in and narrow your prospective IDE list down accordingly. Examples include Ruby, Python, and Java IDE tools.
- Integrations and plugins: With the name integrated development environment, it is no surprise that integrations need to be considered when looking at IDEs. Your IDE is your development portal, so incorporating all your other development tools will improve development workflows and productivity. Poor integrations can cause numerous issues and lead to many headaches.

Image 8: Features of IDE(source: Internet)

- Advantage of IDE: (Aaron Walker, 2021)

  + Serves as a single environment for most, if not all, of a developer's needs, such as version control systems, debugging tools, and Platform-as-a-Service (PaaS).

  + Code completion capabilities improve programming workflow.

+ Automatically checks for errors to ensure top-quality code.

+ Refactoring capabilities allow developers to make comprehensive and mistake-free renaming changes.

+ Maintain a smooth development cycle.

+ Increase developer efficiency and satisfaction.

+ Deliver top-quality software on schedule.

III. Evidences that you have used debugging during the implementation

- I using Debug from my project: Student management

```
1 reference
private void btnUpDate_Click(object sender, EventArgs e)
{
    if (index >= 0)
    {
        lstSinhVien[index].ID = Int32.Parse(txtID.Text);
        lstSinhVien[index].Name = txtName.Text;
        lstSinhVien[index].Gender = txtGender.Text;
        lstSinhVien[index].Age = Int32.Parse(txtAge.Text);
        lstSinhVien[index].Math = double.Parse(txtMath.Text);
        lstSinhVien[index].Chemistry = double.Parse(txtChemistry.Text);
        lstSinhVien[index].Physics = double.Parse(txtPhysics.Text);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;
    }

}
```

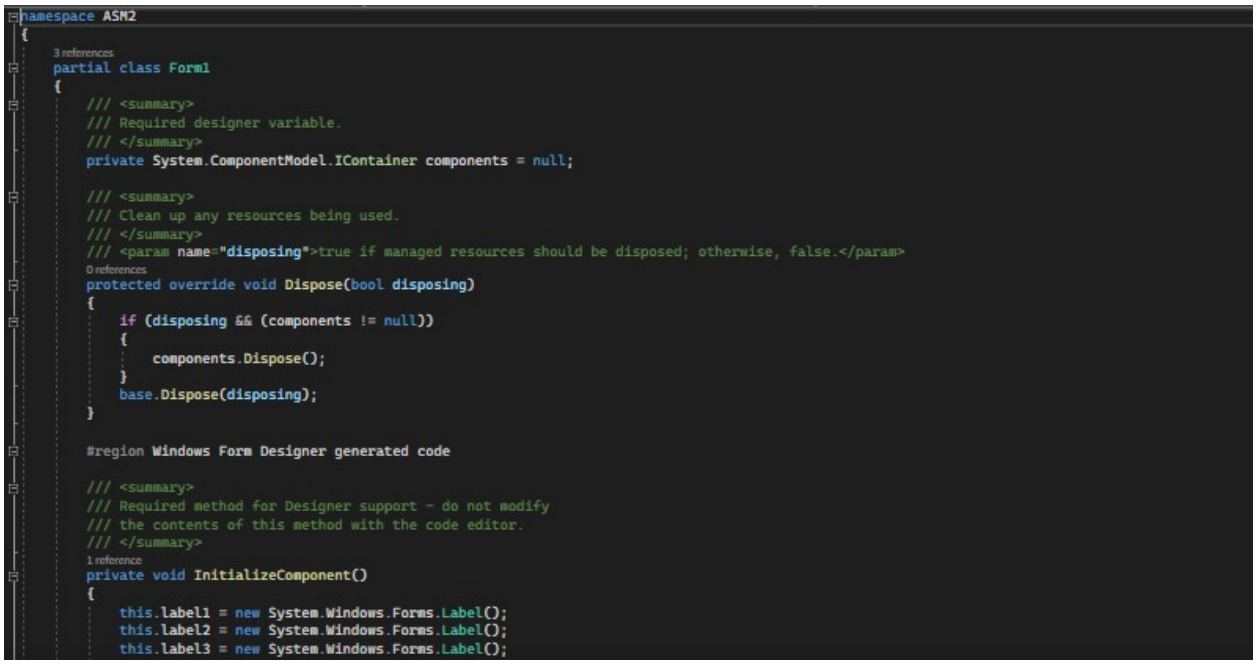
```
output from: Debug
M2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Configuration\v4.0_4.0.0.
M2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Core\v4.0_4.0.0.0__b77a5c
M2.exe' (CLR v4.0.30319: ASM2.exe): Loaded 'C:\WINDOWS\Microsoft.Net\assembly\GAC_MSIL\System.Xml\v4.0_4.0.0.0__b77a5c
program '[12752] ASM2.exe' has exited with code 0 (0x0).
```

Task 4: Design and Implementation

I. Flowchart of the application

II. Source code and screenshots of the final application with explanation

II.1/ Form1.Designer



```
namespace ASM2
{
    3 references
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>
        0 references
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        1 reference
        private void InitializeComponent()
        {
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.label3 = new System.Windows.Forms.Label();
```

```
        private void InitializeComponent()
        {
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.label3 = new System.Windows.Forms.Label();
            this.txtID = new System.Windows.Forms.TextBox();
            this.txtName = new System.Windows.Forms.TextBox();
            this.txtGender = new System.Windows.Forms.TextBox();
            this.btnAdd = new System.Windows.Forms.Button();
            this.btnDeleteID = new System.Windows.Forms.Button();
            this.btnSort = new System.Windows.Forms.Button();
            this.btnUpDate = new System.Windows.Forms.Button();
            this.dataGridView = new System.Windows.Forms.DataGridView();
            this.Column1 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column2 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column3 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column4 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column5 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column6 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.Column7 = new System.Windows.Forms.DataGridViewTextBoxColumn();
            this.txtAge = new System.Windows.Forms.TextBox();
            this.label4 = new System.Windows.Forms.Label();
            this.label5 = new System.Windows.Forms.Label();
            this.label6 = new System.Windows.Forms.Label();
            this.label7 = new System.Windows.Forms.Label();
            this.txtMath = new System.Windows.Forms.TextBox();
            this.txtChemistry = new System.Windows.Forms.TextBox();
            this.txtPhysics = new System.Windows.Forms.TextBox();
            ((System.ComponentModel.ISupportInitialize)(this.dataGridView)).BeginInit();
            this.SuspendLayout();
            //
```

```
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(53, 33);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(18, 13);
this.label1.TabIndex = 0;
this.label1.Text = "ID";
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(53, 73);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(35, 13);
this.label2.TabIndex = 1;
this.label2.Text = "Name";
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(53, 113);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(42, 13);
this.label3.TabIndex = 2;
this.label3.Text = "Gender";
//
// txtID
//
this.txtID.Location = new System.Drawing.Point(149, 26);
this.txtID.Name = "txtID";
this.txtID.Size = new System.Drawing.Size(100, 20);
this.txtID.TabIndex = 3;
//
// txtName
//
this.txtName.Location = new System.Drawing.Point(149, 66);
this.txtName.Name = "txtName";
this.txtName.Size = new System.Drawing.Size(200, 20);
this.txtName.TabIndex = 4;
```

```
// txtGender
//
this.txtGender.Location = new System.Drawing.Point(149, 106);
this.txtGender.Name = "txtGender";
this.txtGender.Size = new System.Drawing.Size(100, 20);
this.txtGender.TabIndex = 5;
//
// btnAdd
//
this.btnAdd.Location = new System.Drawing.Point(56, 197);
this.btnAdd.Name = "btnAdd";
this.btnAdd.Size = new System.Drawing.Size(75, 23);
this.btnAdd.TabIndex = 6;
this.btnAdd.Text = "Add";
this.btnAdd.UseVisualStyleBackColor = true;
this.btnAdd.Click += new System.EventHandler(this.btnAdd_Click);
//
// btnDeleteID
//
this.btnDeleteID.Location = new System.Drawing.Point(160, 197);
this.btnDeleteID.Name = "btnDeleteID";
this.btnDeleteID.Size = new System.Drawing.Size(75, 23);
this.btnDeleteID.TabIndex = 7;
this.btnDeleteID.Text = "DeleteID";
this.btnDeleteID.UseVisualStyleBackColor = true;
this.btnDeleteID.Click += new System.EventHandler(this.btnDeleteID_Click);
//
// btnSort
//
this.btnSort.Location = new System.Drawing.Point(274, 197);
this.btnSort.Name = "btnSort";
this.btnSort.Size = new System.Drawing.Size(75, 23);
this.btnSort.TabIndex = 8;
this.btnSort.Text = "Sort";
this.btnSort.UseVisualStyleBackColor = true;
this.btnSort.Click += new System.EventHandler(this.btnSortByName_Click);
//
// btnUpDate
//
this.btnUpDate.Location = new System.Drawing.Point(397, 197);
this.btnUpDate.Name = "btnUpDate";
this.btnUpDate.Size = new System.Drawing.Size(75, 23);
this.btnUpDate.TabIndex = 9;
this.btnUpDate.Text = "UpDate";
this.btnUpDate.UseVisualStyleBackColor = true;
this.btnUpDate.Click += new System.EventHandler(this.btnUpDate_Click);
//
```

```
//
// dataGridView
//
this.dataGridView.ColumnHeadersHeightSizeMode = System.Windows.Forms.DataGridViewColumnHeadersHeightSizeMode.AutoSize;
this.dataGridView.Columns.AddRange(new System.Windows.Forms.DataGridViewColumn[] {
this.Column1,
this.Column2,
this.Column3,
this.Column4,
this.Column5,
this.Column6,
this.Column7});
this.dataGridView.Location = new System.Drawing.Point(56, 302);
this.dataGridView.Name = "dataGridView";
this.dataGridView.Size = new System.Drawing.Size(484, 154);
this.dataGridView.TabIndex = 10;
this.dataGridView.CellContentClick += new System.Windows.Forms.DataGridViewCellEventHandler(this.dataGridView_CellContentClick);
//
// Column1
//
this.Column1.DataPropertyName = "ID";
this.Column1.HeaderText = "ID";
this.Column1.Name = "Column1";
this.Column1.Width = 50;
//
// Column2
//
this.Column2.DataPropertyName = "Name";
this.Column2.HeaderText = "Name";
this.Column2.Name = "Column2";
this.Column2.Width = 130;
//
// Column3
//
this.Column3.DataPropertyName = "Gender";
this.Column3.HeaderText = "Gender";
this.Column3.Name = "Column3";
this.Column3.Width = 50;
//
// Column4
//
this.Column4.DataPropertyName = "Age";
this.Column4.HeaderText = "Age";
this.Column4.Name = "Column4";
this.Column4.Width = 50;
//
```

```
            //
            // Column5
            //
            this.Column5.DataPropertyName = "Math";
            this.Column5.HeaderText = "Math";
            this.Column5.Name = "Column5";
            this.Column5.Width = 50;
            //
            // Column6
            //
            this.Column6.DataPropertyName = "Chemistry";
            this.Column6.HeaderText = "Chemistry";
            this.Column6.Name = "Column6";
            this.Column6.Width = 60;
            //
            // Column7
            //
            this.Column7.DataPropertyName = "Physics";
            this.Column7.HeaderText = "Physics";
            this.Column7.Name = "Column7";
            this.Column7.Width = 50;
            //
            // txtAge
            //
            this.txtAge.Location = new System.Drawing.Point(149, 154);
            this.txtAge.Name = "txtAge";
            this.txtAge.Size = new System.Drawing.Size(120, 20);
            this.txtAge.TabIndex = 12;
            //
            // label4
            //
            this.label4.AutoSize = true;
            this.label4.Location = new System.Drawing.Point(60, 161);
            this.label4.Name = "label4";
            this.label4.Size = new System.Drawing.Size(26, 13);
            this.label4.TabIndex = 13;
            this.label4.Text = "Age";
            //
            // label5
            //
            this.label5.AutoSize = true;
            this.label5.Location = new System.Drawing.Point(475, 33);
            this.label5.Name = "label5";
            this.label5.Size = new System.Drawing.Size(31, 13);
            this.label5.TabIndex = 14;
            this.label5.Text = "Math";
            //
```

```
            //
            // label6
            //
            this.label6.AutoSize = true;
            this.label6.Location = new System.Drawing.Point(475, 73);
            this.label6.Name = "label6";
            this.label6.Size = new System.Drawing.Size(52, 13);
            this.label6.TabIndex = 15;
            this.label6.Text = "Chemistry";
            //
            // label7
            //
            this.label7.AutoSize = true;
            this.label7.Location = new System.Drawing.Point(475, 113);
            this.label7.Name = "label7";
            this.label7.Size = new System.Drawing.Size(43, 13);
            this.label7.TabIndex = 16;
            this.label7.Text = "Physics";
            //
            // txtMath
            //
            this.txtMath.Location = new System.Drawing.Point(557, 26);
            this.txtMath.Name = "txtMath";
            this.txtMath.Size = new System.Drawing.Size(100, 20);
            this.txtMath.TabIndex = 17;
            //
            // txtChemistry
            //
            this.txtChemistry.Location = new System.Drawing.Point(557, 66);
            this.txtChemistry.Name = "txtChemistry";
            this.txtChemistry.Size = new System.Drawing.Size(100, 20);
            this.txtChemistry.TabIndex = 18;
            //
            // txtPhysics
            //
            this.txtPhysics.Location = new System.Drawing.Point(557, 106);
            this.txtPhysics.Name = "txtPhysics";
            this.txtPhysics.Size = new System.Drawing.Size(100, 20);
            this.txtPhysics.TabIndex = 19;
            //
```

```
            //
            // Form1
            //
            this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
            this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
            this.ClientSize = new System.Drawing.Size(800, 450);
            this.Controls.Add(this.txtPhysics);
            this.Controls.Add(this.txtChemistry);
            this.Controls.Add(this.txtMath);
            this.Controls.Add(this.label7);
            this.Controls.Add(this.label6);
            this.Controls.Add(this.label5);
            this.Controls.Add(this.label4);
            this.Controls.Add(this.txtAge);
            this.Controls.Add(this.dataGridView);
            this.Controls.Add(this.btnUpDate);
            this.Controls.Add(this.btnSort);
            this.Controls.Add(this.btnDeleteID);
            this.Controls.Add(this.btnAdd);
            this.Controls.Add(this.txtGender);
            this.Controls.Add(this.txtName);
            this.Controls.Add(this.txtID);
            this.Controls.Add(this.label3);
            this.Controls.Add(this.label2);
            this.Controls.Add(this.label1);
            this.Name = "Form1";
            this.Text = "Form1";
            ((System.ComponentModel.ISupportInitialize)(this.dataGridView)).EndInit();
            this.ResumeLayout(false);
            this.PerformLayout();

        }
```

```
        #endregion

        private System.Windows.Forms.Label label1;
        private System.Windows.Forms.Label label2;
        private System.Windows.Forms.Label label3;
        private System.Windows.Forms.TextBox txtID;
        private System.Windows.Forms.TextBox txtName;
        private System.Windows.Forms.TextBox txtGender;
        private System.Windows.Forms.Button btnAdd;
        private System.Windows.Forms.Button btnDeleteID;
        private System.Windows.Forms.Button btnSort;
        private System.Windows.Forms.Button btnUpDate;
        private System.Windows.Forms.DataGridView dataGridView;
        private System.Windows.Forms.TextBox txtAge;
        private System.Windows.Forms.Label label4;
        private System.Windows.Forms.Label label5;
        private System.Windows.Forms.Label label6;
        private System.Windows.Forms.Label label7;
        private System.Windows.Forms.TextBox txtMath;
        private System.Windows.Forms.TextBox txtChemistry;
        private System.Windows.Forms.TextBox txtPhysics;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column1;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column2;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column3;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column4;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column5;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column6;
        private System.Windows.Forms.DataGridViewTextBoxColumn Column7;
    }
}
```

II.2/ Form1.cs

- Class Form 1:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ASM2
{
    3 references
    public partial class Form1 : Form
    {
        List<SinhVien> lstSinhVien;
        int index;
        1 reference
        public Form1()
        {
            InitializeComponent();
            lstSinhVien = new List<SinhVien>();
        }

        1 reference
        public bool CheckInput()
        {

            if (string.IsNullOrWhiteSpace(txtID.Text))
            {
                MessageBox.Show("You have not entered the Student Id !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtID.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtName.Text))
            {
                MessageBox.Show("You have not entered the Student Name !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtName.Focus();
                return false;
            }
            if (string.IsNullOrWhiteSpace(txtGender.Text))
            {
                MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
                txtGender.Focus();
                return false;
```

```
        if (string.IsNullOrWhiteSpace(txtAge.Text))
        {
            MessageBox.Show("You have not entered the Student Id !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
            txtAge.Focus();
            return false;
        }
        if (string.IsNullOrWhiteSpace(txtMath.Text))
        {
            MessageBox.Show("You have not entered the Student Name !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
            txtMath.Focus();
            return false;
        }
        if (string.IsNullOrWhiteSpace(txtChemistry.Text))
        {
            MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
            txtChemistry.Focus();
            return false;
        }
        if (string.IsNullOrWhiteSpace(txtPhysics.Text))
        {
            MessageBox.Show("You have not entered the Student Mark !", "Notification", MessageBoxButtons.OK, MessageBoxIcon.Error);
            txtPhysics.Focus();
            return false;
        }
        return true;
```

- Class SinhVien:
  Helps to define and store student attributes

```
6 references
class SinhVien
{
    4 references
    public int ID { get; set; }
    6 references
    public string Name { get; set; }
    4 references
    public string Gender { get; set; }
    4 references
    public int Age { get; set; }
    4 references
    public double Math { get; set; }
    4 references
    public double Chemistry { get; set; }
    4 references
    public double Physics { get; set; }
    0 references
    public SinhVien()
    {
        ID = 0;
        Name = "No name";
        Gender = "No gender";
        Age = 0;
        Math = 0;
        Chemistry = 0;
        Physics = 0;
    }
    1 reference
    public SinhVien(int svID, string svName, string svGender, int svAge, double svMath, double svChemistry, double svPhysics)
    {
        ID = svID;
        Name = svName;
        Gender = svGender;
        Age = svAge;
        Math = svMath;
        Chemistry = svChemistry;
        Physics = svPhysics;
    }
}
}
```

- private void dataGridView_CellContentClick:

  It supports input data to be filled in to the correct

```
1 reference
private void dataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    index = e.RowIndex;
    if (index >= 0)
    {
        txtID.Text = lstSinhVien[index].ID.ToString();
        txtName.Text = lstSinhVien[index].Name;
        txtGender.Text = lstSinhVien[index].Gender;
        txtAge.Text = lstSinhVien[index].Age.ToString();
        txtMath.Text = lstSinhVien[index].Math.ToString();
        txtChemistry.Text = lstSinhVien[index].Chemistry.ToString();
        txtPhysics.Text = lstSinhVien[index].Physics.ToString();
    }
}
```

- Add_Click:

```
1 reference
private void btnAdd_Click(object sender, EventArgs e)
{
    if (CheckInput())
    {
        int svID;
        string svName;
        string svGender;
        int svAge;
        double svMath;
        double svChemistry;
        double svPhysics;
        svID = Int32.Parse(txtID.Text);
        svName = txtName.Text;
        svGender = txtGender.Text;
        svAge = Int32.Parse(txtAge.Text);
        svMath = double.Parse(txtMath.Text);
        svChemistry = double.Parse(txtChemistry.Text);
        svPhysics = double.Parse(txtPhysics.Text);
        SinhVien sv = new SinhVien(svID, svName, svGender, svAge, svMath, svChemistry, svPhysics);
        lstSinhVien.Add(sv);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;
        dataGridView.Refresh();

    }
}
```

- Delete_Click

```
1 reference
private void btnDeleteID_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Do you want to delete?", "Notification", MessageBoxButtons.OKCancel, MessageBoxIcon.Warning) == DialogResult.OK)
    {
        lstSinhVien.RemoveAt(index);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;

    }
}
```

- Update_Click

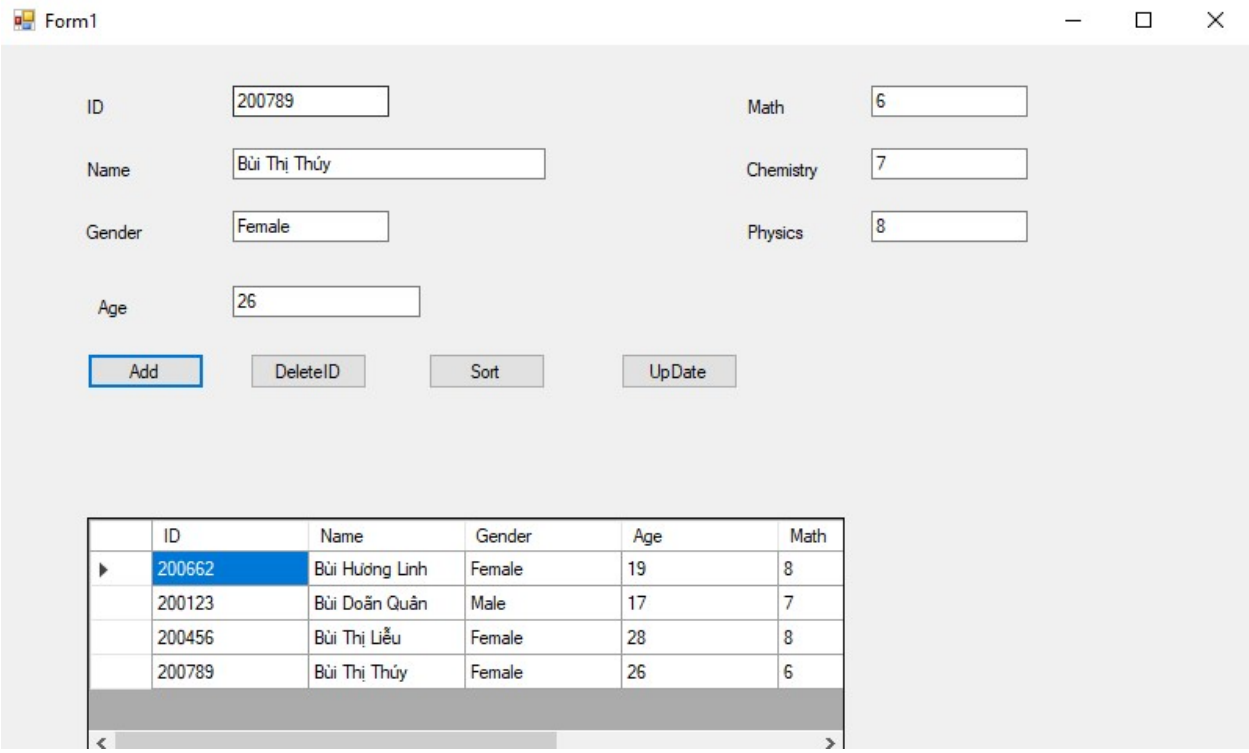
```
1 reference
private void btnUpDate_Click(object sender, EventArgs e)
{
    if (index >= 0)
    {
        lstSinhVien[index].ID = Int32.Parse(txtID.Text);
        lstSinhVien[index].Name = txtName.Text;
        lstSinhVien[index].Gender = txtGender.Text;
        lstSinhVien[index].Age = Int32.Parse(txtAge.Text);
        lstSinhVien[index].Math = double.Parse(txtMath.Text);
        lstSinhVien[index].Chemistry = double.Parse(txtChemistry.Text);
        lstSinhVien[index].Physics = double.Parse(txtPhysics.Text);
        dataGridView.DataSource = null;
        dataGridView.DataSource = lstSinhVien;
    }

}
```

- Sort_Click

```
1 reference
private void btnSort_Click(object sender, EventArgs e)
{
    lstSinhVien.Sort((a, b) => a.Name.CompareTo(b.Name));
    dataGridView.DataSource = null;
    dataGridView.DataSource = lstSinhVien;
}
```

II.3/ Form1.cs[Design]

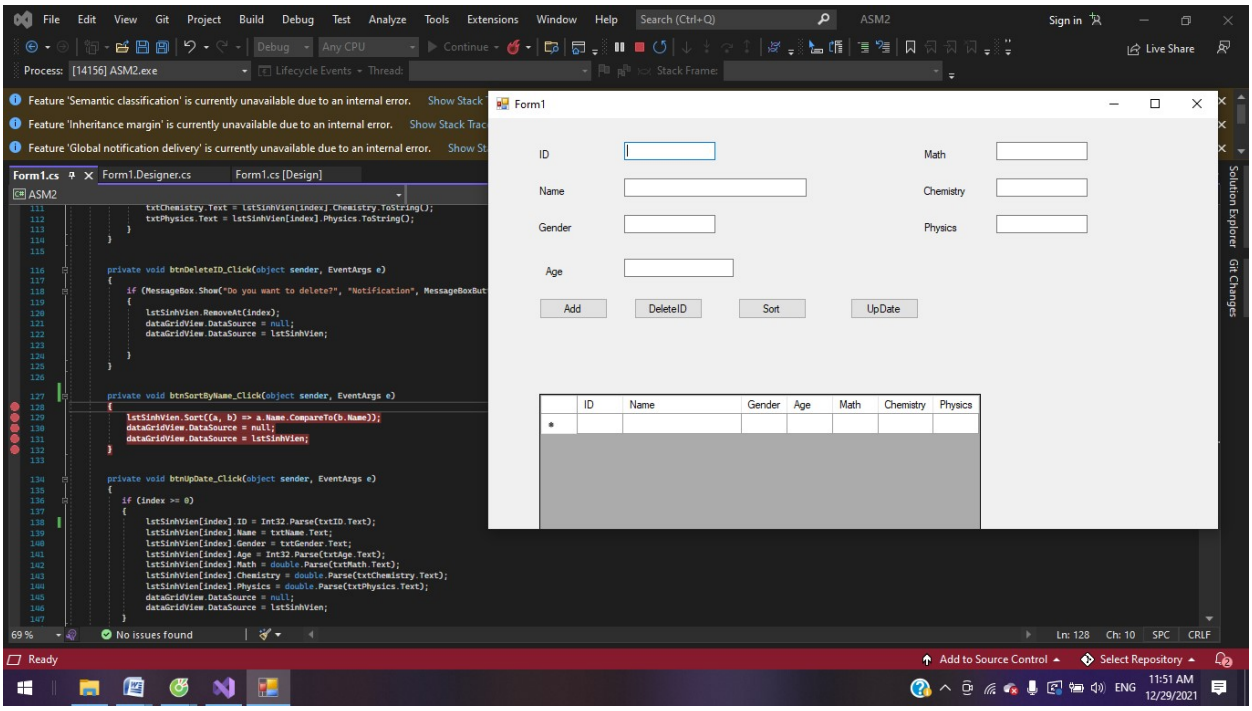


III. Debug and Test plan

III.1/ Debug

- Chose line debug and start with F5

```
127        private void btnSortByName_Click(object sender, EventArgs e)
128        {
129            lstSinhVien.Sort((a, b) => a.Name.CompareTo(b.Name));
130            dataGridView.DataSource = null;
131            dataGridView.DataSource = lstSinhVien;
132        }
133
```



Bibliography

(2021, December 21). Retrieved from Learn Computer Science: https://www.learncomputerscienceonline.com/procedural-programming/?fbclid=IwAR0hkIuXne4jhKyGuapfIGTLkAS4K5bdtplmrmt93AzP5rSAGGaqiVIy6wU

Cordrey, L. (2021, December 21). Retrieved from popularask.net: https://popularask.net/which-of-the-following-is-a-main-characteristic-of-procedural-programming-language/?fbclid=IwAR1EOdIZqEnRmfwDA15-V1zZHECGGma92dS13gFxfN8_yiPUP5unXvtaX8M

Doherty, E. (2021, December 21). Retrieved from educative.io: https://www.educative.io/blog/object-oriented-programming?fbclid=IwAR1m0gxx1AGUz5O9ShCBWmfY6WfCKvM2fa_qE6FzGZ4aK_WSSCCkG2bTmls

Techopedia, C. (2021, December 21). Retrieved from techopedia.com: https://www.techopedia.com/definition/7083/event-driven-program?fbclid=IwAR0HS5LQXS3e0mkH5zPizLc2G1d9CoE0_VmKavQD8xpOWXJld_KsYWWB4P8

Walker, A. (2021, December 21). Retrieved from g2.com: https://www.g2.com/articles/ide?fbclid=IwAR0irCNIu44W2MPaxuz0uuFOVmJeXZLY2Fb6AkbFoC-FVytGFkLCR7-UN9U