

**ข้อกำหนด:** ให้นักศึกษาเขียนโปรแกรมภาษาซีตามที่โจทย์กำหนด แล้วบันทึกเป็นไฟล์ .c ให้ตั้งชื่อว่า cs102Hw4-x-y เมื่อ x คือ เลขข้อ และ y คือเลขทะเบียนของนักศึกษา (ตัวอย่าง นักศึกษาเลขทะเบียน 6709670000 จะต้องตั้งชื่อไฟล์คำตอบข้อสอบของการบ้าน 4 ข้อ ที่ 1 ว่า cs102Hw4-1-6709670000.c) แล้วอัปโหลดไฟล์ขึ้นกล่องส่งในคอร์สเว็บให้ตรงข้อ **ไม่เกินกำหนดส่ง**

**กำหนดส่ง:** การบ้านครั้งที่ 4 ข้อ 1-3 ไม่เกิน 23:55 ของวันศุกร์ที่ 29 พฤศจิกายน 2567

การบ้านครั้งที่ 4 ข้อ 4 ไม่เกิน 23:55 ของวันจันทร์ที่ 2 ธันวาคม 2567

1. เขียนโปรแกรมเพื่อรับค่าตัวเลข n จำนวนจากผู้ใช้และเก็บในอาร์เรย์ จากนั้นให้ทำการสลับค่าของตัวเลขในอาร์เรย์เป็นคู่ โดยใช้ฟังก์ชัน swap\_elements() ซึ่งทำการสลับค่าของสมาชิกในอาร์เรย์ตามตำแหน่งที่กำหนดไว้ โดยส่งข้อมูลแบบ pass by reference ซึ่งหมายความว่า การสลับจะเกิดขึ้นกับค่าจริงที่เก็บในอาร์เรย์ โดยโปรแกรมนี้นี้จะให้ผู้ใช้กำหนดจำนวน n ของตัวเลขที่ต้องการป้อน จากนั้นโปรแกรมจะขอให้ผู้ใช้ระบุตำแหน่งของตัวเลขสองตำแหน่งที่ต้องการสลับกัน โปรแกรมจะเรียกฟังก์ชัน swap\_elements เพื่อทำการสลับค่าที่ตำแหน่งนั้นและแสดงผลที่ได้ออกมา

โดยกำหนดให้ใช้ฟังก์ชันต้นแบบ (function prototype ดังนี้ void swap\_elements(int \*arr, int pos1, int pos2);

หมายเหตุ การทำงานหลักจะเป็นไปตามตัวอย่างการรันด้านล่าง นอกจากนี้ นักศึกษาสามารถเพิ่มเติมส่วนของการทำงานที่จำเป็นได้ตามความเหมาะสม

**ตัวอย่างการรันครั้งที่ 1** (ข้อมูลเข้าไฮไลต์ด้วยสีเหลือง)

Enter the number of elements (max 100): 4

Enter 4 numbers: 5 15 25 35

Enter first position to swap (0-3): 0

Enter second position to swap (0-3): 2

After swapping elements at positions 0 and 2:

25 15 5 35

**ตัวอย่างการรันครั้งที่ 2** (ข้อมูลเข้าไฮไลต์ด้วยสีเหลือง)

Enter the number of elements (max 100): 5

Enter 5 numbers: 10 20 30 40 50

Enter first position to swap (0-4): 1

Enter second position to swap (0-4): 3

After swapping elements at positions 1 and 3:

10 40 30 20 50

2. เขียนโปรแกรมเพื่อเก็บข้อมูลลงทะเบียนคนยากจน โดยใช้โครงสร้างข้อมูล (struct) เพื่อเก็บรายละเอียดของผู้ลงทะเบียนแต่ละคน และใช้ array ในการจัดเก็บข้อมูลผู้ลงทะเบียนหลาย ๆ คน โปรแกรมจะให้ผู้ใช้งานสามารถเพิ่มข้อมูลผู้ลงทะเบียน และแสดงรายชื่อผู้ลงทะเบียนทั้งหมดที่อยู่ในระบบ

กำหนดให้ใช้โครงสร้าง struct สำหรับเก็บข้อมูลผู้ลงทะเบียน โดยให้ข้อมูลประกอบด้วย:

- id รหัสผู้ลงทะเบียน (ชนิด int)
- name ชื่อผู้ลงทะเบียน (ชนิด char array ขนาด 50)
- age อายุของผู้ลงทะเบียน (ชนิด int)
- income รายได้ต่อเดือน (ชนิด float)

และ กำหนดให้ array ของ struct มีขนาดสูงสุด 100 เพื่อเก็บข้อมูลผู้ลงทะเบียนหลายคนได้

โดยในฟังก์ชัน main() ให้มีเมนูเพื่อให้ผู้ใช้งานสามารถเลือกทำงานได้ตามต้องการ ได้แก่ 1. การเพิ่มข้อมูลผู้ลงทะเบียนใหม่ 2. การแสดงข้อมูลผู้ลงทะเบียนทั้งหมด และ 3. การออกจากโปรแกรม สำหรับการเพิ่มข้อมูลผู้ลงทะเบียนใหม่ ผู้ใช้จะต้องใส่ข้อมูลทั้งหมดใน struct เช่น รหัส ชื่อ อายุ และรายได้ต่อเดือน สำหรับการแสดงข้อมูลของผู้ลงทะเบียนทั้งหมดในระบบให้แสดงผลในรูปแบบตารางที่อ่านง่าย

หมายเหตุ การทำงานหลักจะเป็นไปตามตัวอย่างการรันด้านล่าง นอกจากนี้ก็ยังสามารถเพิ่มเติมส่วนของการทำงานที่จำเป็นได้ตามความเหมาะสม

**ตัวอย่างการรันครั้งที่ 1** (ข้อมูลเข้าโหลต์ด้วยสีเหลือง)

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

==

Enter your choice: 2

No registered persons to display.

==

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

==

Enter your choice: 3

Exiting program.

## ตัวอย่างการรันครั้งที่ 2 (ข้อมูลเข้าไอลท์ด้วยสีเหลือง)

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

==

Enter your choice: 1

Enter ID: 101

Enter Name: John Doe

Enter Age: 45

Enter Monthly Income: 2000.00

Person registered successfully.

==

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

==

Enter your choice: 1

Enter ID: 102

Enter Name: Jane Smith

Enter Age: 32

Enter Monthly Income: 1500.00

Person registered successfully.

==

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

==

Enter your choice: 2

ID	Name	Age	Monthly Income
101	John Doe	45	2000.00
102	Jane Smith	32	1500.00

==

Poverty Registration System

1. Register New Person
2. Display All Registered Persons
3. Exit

Enter your choice: 3

Exiting program.

3. เขียนโปรแกรม เพื่อจัดการข้อมูลลูกค้าธนาคาร โดยใช้โครงสร้าง struct เพื่อเก็บข้อมูลของลูกค้าแต่ละคน และใช้ array เพื่อจัดเก็บข้อมูลลูกค้าหลาย ๆ คน โดยโปรแกรมนี้อาจช่วยในการเก็บข้อมูลพื้นฐานของลูกค้าแต่ละคน เช่น หมายเลขบัญชี ชื่อลูกค้า และยอดเงินในบัญชี และโปรแกรมสามารถแสดงรายงานของลูกค้าทั้งหมดในธนาคารและมีฟังก์ชันสำหรับการฝากและถอนเงินจากบัญชีของลูกค้า

กำหนดให้สร้างโครงสร้าง struct สำหรับเก็บข้อมูลลูกค้า โดยให้ข้อมูลมีดังนี้:

- accountNumber หมายเลขบัญชี (ชนิด int)
- name ชื่อลูกค้า (ชนิด char array ขนาด 50)
- balance ยอดเงินในบัญชี (ชนิด float)

และกำหนดให้มีฟังก์ชันต่าง ๆ ดังต่อไปนี้

- ฟังก์ชัน addCustomer() สำหรับเพิ่มข้อมูลลูกค้าใหม่
- ฟังก์ชัน deposit() สำหรับฝากเงินเข้าบัญชี โดยระบุหมายเลขบัญชีและจำนวนเงินที่ต้องการฝาก
- ฟังก์ชัน withdraw() สำหรับถอนเงินออกจากบัญชี โดยระบุหมายเลขบัญชีและจำนวนเงินที่ต้องการถอน
- ฟังก์ชัน displayCustomers() เพื่อแสดงรายงานของลูกค้าทั้งหมดในธนาคาร

หมายเหตุ การทำงานหลักจะเป็นไปตามตัวอย่างการรันด้านล่าง นอกจากนั้นนักศึกษาสามารถเพิ่มเติมส่วนของการทำงานที่จำเป็นได้ตามความเหมาะสม

**ตัวอย่างการรันครั้งที่ 1** (ข้อมูลเข้าใส่ไลต์ด้วยสีเหลือง)

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 1

Enter account number: 1001

Enter customer name: Alice

Enter initial balance: 1500.00

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 4

Account Number: 1001, Name: Alice, Balance: 1500.00

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 5

Exiting program.

## ตัวอย่างการรันครั้งที่ 2 (ข้อมูลเข้าไอล็อตด้วยสีเหลือง)

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 1

Enter account number: 1001

Enter customer name: Alice

Enter initial balance: 1500.00

Customer added successfully.

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 1

Enter account number: 1002

Enter customer name: Bob

Enter initial balance: 2000.00

Customer added successfully.

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 2

Enter account number for deposit: 1001

Enter amount to deposit: 500.00

Deposit successful. New balance: 2000.00

Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 3

Enter account number for withdrawal: 1002

Enter amount to withdraw: 700.00

Withdrawal successful. New balance: 1300.00

## Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 4

Customer List:

Account Number: 1001, Name: Alice, Balance: 2000.00

Account Number: 1002, Name: Bob, Balance: 1300.00

## Bank Management System

1. Add New Customer
2. Deposit Money
3. Withdraw Money
4. Display All Customers
5. Exit

Enter your choice: 5

Exiting program.

4. เขียนโปรแกรมเพื่อสร้างระบบลงทะเบียนเรียนรายวิชาของนักศึกษามหาวิทยาลัยธรรมศาสตร์ โดยโปรแกรมจะใช้ struct สำหรับเก็บข้อมูลรายวิชาและข้อมูลนักศึกษา ใช้ function ในการจัดการข้อมูล และใช้ file ในการบันทึกและดึงข้อมูลการลงทะเบียนรายวิชา โดยโปรแกรมนี้สามารถ 1. เพิ่มข้อมูลรายวิชาใหม่ 2. เพิ่มข้อมูลนักศึกษาใหม่ 3. ลงทะเบียนรายวิชาให้กับนักศึกษา 4. แสดงรายวิชาที่นักศึกษาได้ลงทะเบียนไว้ 5. บันทึกข้อมูลการลงทะเบียนลงในไฟล์เพื่อให้สามารถเรียกใช้ได้เมื่อเปิดโปรแกรมใหม่ 6. อ่านข้อมูลการลงทะเบียนลงจากไฟล์ และ 7. ออกจากโปรแกรม

กำหนดให้สร้างโครงสร้าง struct สำหรับเก็บข้อมูลลูกค้า โดยให้ข้อมูลมีดังนี้:

โครงสร้าง Course: สำหรับเก็บข้อมูลรายวิชา ประกอบด้วย

- courseCode รหัสวิชา (ชนิด char array ขนาด 10)
- courseName ชื่อวิชา (ชนิด char array ขนาด 50)
- credits จำนวนหน่วยกิต (ชนิด int)

โครงสร้าง Student: สำหรับเก็บข้อมูลนักศึกษาและรายวิชาที่ลงทะเบียนไว้ ประกอบด้วย

- studentID รหัสนักศึกษา (ชนิด char array ขนาด 10)
- name ชื่อนักศึกษา (ชนิด char array ขนาด 50)
- registeredCourses รายวิชาที่นักศึกษาลงทะเบียน (ใช้ array ของ Course ขนาดสูงสุด 10 รายวิชา)
- courseCount จำนวนรายวิชาที่ลงทะเบียน (ชนิด int)

กำหนดให้สร้างฟังก์ชันสำหรับการจัดการข้อมูลดังต่อไปนี้

- ฟังก์ชัน addCourse() สำหรับเพิ่มข้อมูลรายวิชาใหม่
- ฟังก์ชัน registerCourse() สำหรับลงทะเบียนรายวิชาให้กับนักศึกษา โดยใช้รหัสวิชาในการค้นหาวิชาจาก array ของรายวิชาที่มีอยู่
- ฟังก์ชัน displayStudentCourses() เพื่อแสดงรายวิชาที่นักศึกษาได้ลงทะเบียนไว้
- ฟังก์ชัน saveData() สำหรับบันทึกข้อมูลการลงทะเบียนในไฟล์
- ฟังก์ชัน loadData() สำหรับอ่านข้อมูลการลงทะเบียนจากไฟล์
- ในฟังก์ชัน main() ให้โปรแกรมมีเมนูสำหรับให้ผู้ใช้สามารถเลือกทำงานต่าง ๆ ได้

และกำหนดให้บันทึกข้อมูลนักศึกษาและรายวิชาในไฟล์ “registration\_data.txt” โดยในโปรแกรมสามารถอ่านข้อมูลจากไฟล์นี้เพื่อโหลดการลงทะเบียนที่มีอยู่ก่อนหน้านี้ได้ โดยตัวอย่างการรันครั้งที่ 1 และ 2 เป็นตัวอย่างการรันโปรแกรมต่อเนื่องกัน

หมายเหตุ การทำงานหลักจะเป็นไปตามตัวอย่างการรันด้านล่าง นอกจากนี้นักศึกษาสามารถเพิ่มเติมส่วนของการทำงานที่จำเป็นได้ตามความเหมาะสม

**ตัวอย่างการรันครั้งที่ 1** (ข้อมูลเข้าไอล็อตด้วยสีเหลือง)

==== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 1

Enter Course Code: CS101

Enter Course Name: Introduction to Computer Science

Enter Credits: 3

Course added successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 1

Enter Course Code: MA101

Enter Course Name: Calculus I

Enter Credits: 4

Course added successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 2

Enter Student ID: S001

Enter Student Name: Alice

Student added successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 3

Enter Student ID: S001

Enter Course Code to Register: CS101

Course registered successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 3

Enter Student ID: S001



Enter Course Code to Register: MA101

Course registered successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 4

Enter Student ID: S001

Courses registered by Alice:

Course Code	Course Name	Credits
CS101	Introduction to Computer Science	3
MA101	Calculus I	4

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 5

Data saved successfully.

===== Course Registration System =====

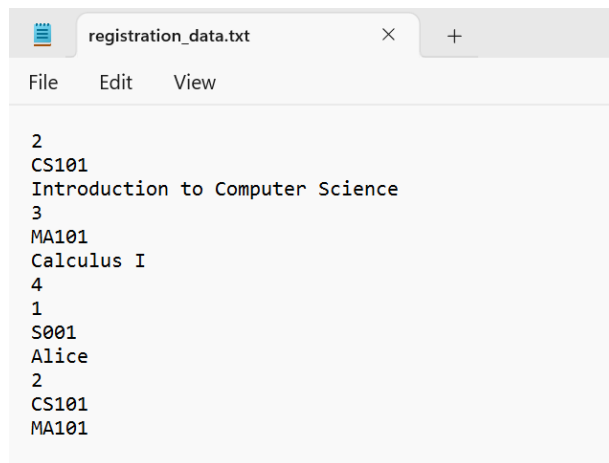
1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 7

Data saved successfully.

Exiting the program. Goodbye!

ตัวอย่างไฟล์ “registration\_data.txt” หลังการรันครั้งที่ 1



```
2
CS101
Introduction to Computer Science
3
MA101
Calculus I
4
1
S001
Alice
2
CS101
MA101
```

คำอธิบายเนื้อหาในไฟล์ “registration\_data.txt” มีความหมายดังต่อไปนี้

จำนวนรายวิชา: 2

รายละเอียดรายวิชา:

รายวิชาที่ 1:

รหัสวิชา: CS101

ชื่อวิชา: Introduction to Computer Science

หน่วยกิต: 3

รายวิชาที่ 2:

รหัสวิชา: MA101

ชื่อวิชา: Calculus I

หน่วยกิต: 4

จำนวนนักศึกษา: 1

รายละเอียดนักศึกษา:

รหัสนักศึกษา: S001

ชื่อนักศึกษา: Alice

จำนวนรายวิชาที่ลงทะเบียน: 2

รหัสวิชาที่ลงทะเบียน:

CS101

MA101

## ตัวอย่างการรันครั้งที่ 2 (ข้อมูลเข้าไอล็อตด้วยสีเหลือง)

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 6

Data loaded successfully.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 4

Enter Student ID: S001

Courses registered by Alice:

Course Code	Course Name	Credits
CS101	Introduction to Computer Science	3
MA101	Calculus I	4

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 3

Enter Student ID: S001

Enter Course Code to Register: PH101

Course Code not found.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 3

Enter Student ID: S999

Student ID not found.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 3

Enter Student ID: S001

Enter Course Code to Register: CS101

Course already registered by the student.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 4

Enter Student ID: S002

Student ID not found.

===== Course Registration System =====

1. Add New Course
2. Add New Student
3. Register Course for Student
4. Display Student's Registered Courses
5. Save Data
6. Load Data
7. Exit

Select an option (1-7): 7

Data saved successfully.

Exiting the program. Goodbye!