International University

School of Electrical Engineering

Programming for Engineers Laboratory

EE058IU

Variable – Data Types – Making Decisions

Submitted by

Member: Nguyễn Thành Danh – EEACIU24018

Member: Nguyễn Huyền Đồng – EEACIU24023

Date Performed:	Oct 16 2025
Date Submitted:	Oct 24 2025
Lab Section:	Lab 1
Course Instructor:	M. Eng Nguyen Minh Thien

GRADING CHECKLIST

Number	Content	Satisfied?		Score	Comment
1	**Format (max 9%)**				
	- Font type	Yes	No		
	- Font size	Yes	No		
	- Lab title	Yes	No		
	- Page number	Yes	No		
	- Table of contents	Yes	No		
	- Header/Footer	Yes	No		
	- List of figures (if exists)	Yes	No		
	- List of tables (if exists)	Yes	No		
	- Lab report structure	Yes	No		
2	**English Grammar and Spelling (max 6%)**				
	- Grammar	Yes	No		
	- Spelling	Yes	No		
3	**Data and Result Analysis (max 85%)**				
	Total Score				

Date:

Signature

TEAM CONTRIBUTION

Task	EEACIU24018	EEACIU24023
Collaboration	Find solution	Find solution, cordinate task
Lab preparation and experiment	Convert solution to code, prepare data for testing	Record results, check error
Report writing	Write results and conclusion/discussion	Write experiment procedure

Table of Contents

List of Figures

List of Tables

Discussion of Fundamentals

If Selection Statement

Selection statements choose among alternative courses of action. For example, suppose the passing grade on an exam is 60. The following pseudocode statement determines whether the condition "student's grade is greater than or equal to 60" is *true* or *false*:

> If student's grade is greater than or equal to 60
>   Print "Passed"

If *true*, then "Passed" is printed, and the next pseudocode statement in order is "performed." Remember that pseudocode isn't a real programming language. If *false*, the printing is ignored, and the next pseudocode statement in order is performed.

The preceding pseudocode is written in C as

```c
if (grade >= 60) {
    puts("Passed");
} // end if
```

Variables and Data types

Variables are names for locations in the computer's memory where data is stored. In C programming, you must specify a data type for every variable, which tells the computer what kind of data it will hold (e.g., a whole number, a decimal number, a character, etc.).

Switch-case Statement

The switch-case statement is a control flow statement used to select one of many code blocks to be executed. It provides a cleaner and more readable alternative to a long series of if-else if-else statements, especially when you are checking a single variable against multiple constant values.

While Statement

The while statement in C is a fundamental looping construct used to repeatedly execute a block of code as long as a given condition remains true. It's often called a pre-test loop because it checks the condition before running the code inside the loop.

Experimental Procedure

Problem 1: Check number is even or odd

START

    PRINT "Input an integer:"

    READ number

    IF number % 2 == 0 THEN
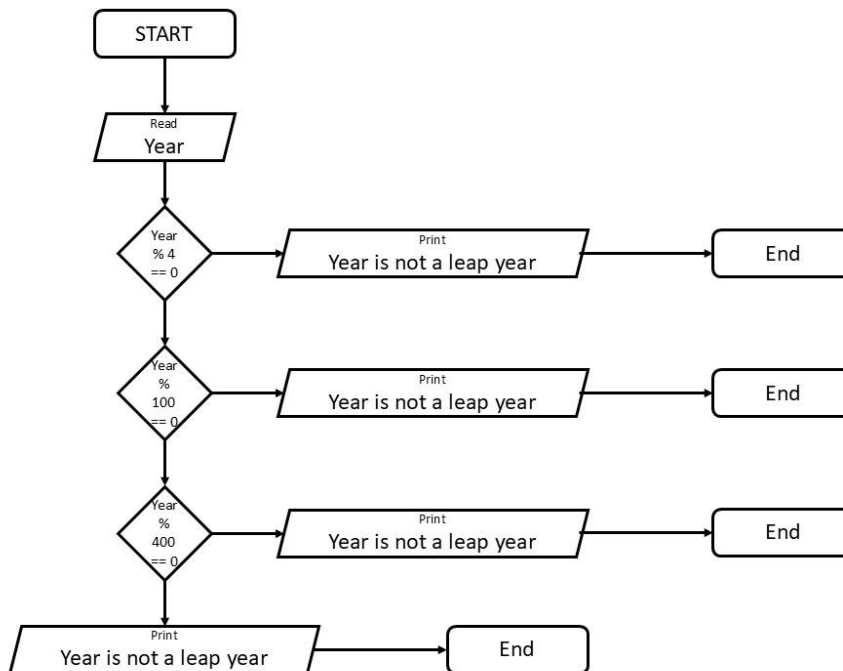
        PRINT number, "is an even integer"

    ELSE

        PRINT number, "is an odd integer"

    END IF

END

Problem 2: Check year is leap or not

Problem 3: Check type of triangle

START

    PRINT "Input three sides of triangle:"

    READ side1, side2, side3

    IF (side1 + side2 > side3) AND (side1 + side3 > side2) AND (side2 + side3 > side1)

```
        IF (side1 == side2) AND (side2 == side3) THEN
            PRINT "This is an equilateral triangle."
        ELSE IF (side1 == side2) OR (side2 == side3) OR (side1 == side3) THEN
            PRINT "This is an isosceles triangle."
        ELSE
            PRINT "This is a scalene triangle."
        END IF
    ELSE
        PRINT "This is not a triangle."
END
```

Problem 4: Calculate diameter, circumference, area of circle with radius and unit

```
START
    PRINT "Enter measurement unit:"
    READ unit
    PRINT "Enter the radius of a circle:"
    READ radius
    diameter = 2 * radius
    circumference = 2 * 3.1416 * radius
    area = 3.1416 * radius * radius
    PRINT "Diameter of circle =", diameter, unit
    PRINT "Circumference of circle =", circumference, unit
    PRINT "Area of circle =", area, "sq.", unit
END
```

Problem 5: Find roots of quadratic equations

```
START
    PRINT "Enter coefficients a, b, and c:"
    READ a, b, c
    COMPUTE discriminant = b*b - 4*a*c
    IF discriminant > 0 THEN
```

```
    root1 = (-b + sqrt(discriminant)) / (2*a)

    root2 = (-b - sqrt(discriminant)) / (2*a)

    PRINT "Roots are real and different."

    PRINT "Root 1 =", root1

    PRINT "Root 2 =", root2

  ELSE IF discriminant == 0 THEN

    root = -b / (2*a)

    PRINT "Roots are real and equal."

    PRINT "Root 1 = Root 2 =", root

  ELSE

    PRINT "No real roots. The equation has complex roots."

  END IF

END
```

Problem 6. Calculate the price passenger has to pay for distance

```
START

  PRINT "Enter total distance in km:"

  READ distance

  IF distance <= 1 THEN

    total_cost = 10000

  ELSE IF distance > 1 AND distance <= 30 THEN

    total_cost = 10000 + (distance - 1) * 5000

  ELSE

    total_cost = 10000 + (29 * 5000) + (distance - 30) * 3000

  END IF

  PRINT "The total cost that a passenger has to pay:", total_cost, "VND"

END
```

Problem 7

```
START

  PRINT "Input the Name of the Student:"
```

```
        READ name
        PRINT "Input the No of the student:"
        READ id
        PRINT "Input the marks of Literature, Math and English:"
        READ literature, math, english
        total = literature + math + english
        average = total / 3.0
        PRINT "Name of Student:", name
        PRINT "ID:", id
        PRINT "Marks in Literature:", literature
        PRINT "Marks in Math:", math
        PRINT "Marks in English:", english
        PRINT "Total Marks =", total
        PRINT "Average =", average
        IF average >= 60 THEN
            PRINT "Division: A"
        ELSE IF average >= 48 AND average < 60 THEN
            PRINT "Division: B"
        ELSE IF average >= 36 AND average < 48 THEN
            PRINT "Division: Pass"
        ELSE
            PRINT "Division: Fail"
        END IF
END
```

Bonus 1

```
START
    REPEAT
        PRINT "Enter account number (-1 to end):"
        READ account_number
```

```
    IF account_number == -1 THEN
        EXIT LOOP
    END IF
    PRINT "Enter beginning balance:"
    READ beginning_balance
    PRINT "Enter total charges:"
    READ total_charges
    PRINT "Enter total credits:"
    READ total_credits
    PRINT "Enter credit limit:"
    READ credit_limit
    new_balance = beginning_balance + total_charges - total_credits
    PRINT "Account:", account_number
    PRINT "Credit limit:", credit_limit
    PRINT "Balance:", new_balance
    IF new_balance > credit_limit THEN
        PRINT "Credit Limit Exceeded."
    END IF
  UNTIL account_number == -1
END
```

Bonus 2

```
START
  PRINT "Enter a five-digit integer:"
  READ number
  tmp = number
  WHILE tmp != 0
    reminder = tmp % 10
    reversed_number = reversed_number * 10 + reminder
    tmp /= 10
  END_WHILE
```

```
  IF (reversed_number == number) THEN
    PRINT "Number is a palindrome"
END
```

Bonus 3

```
START
  PRINT "Enter binary value: "
  READ number
  base = 1, decimal = 0
  WHILE number != 0
    reminder = number % 10
    number /= 10
    decimal += reminder * base
    base *= 2
  END_WHILE
END
```

Experimental Results

Problem 1:
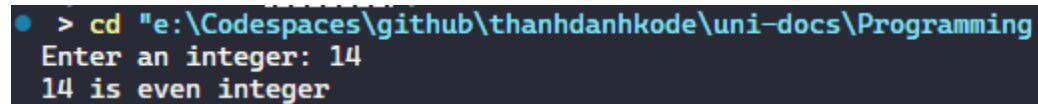
```c
#include <stdio.h>

/**
 *
 * This program print the input number is a even or odd integer
 *
 */
int main()
{
  int num;
  printf("Enter an integer: "); // Prompt user to input a number
  scanf("%d", &num);

  num % 2 == 0 ? printf("%d is even integer\n", num) : printf("%d is odd integer\n",
num);

  return 0;
}
```

Result

Figure 1. Result of the program in Problem 1 – Even/odd number checking.

Problem 2:

```c
#include <stdio.h>
/**
 *
 * This program check the input year is a leap year or not
 *
 */
int main()
{
  int year;

  printf("Input a year: "); // Prompt user to input a year
  scanf("%d", &year);        // Get year from user

  // Condition block for checking input year is a leap year
  if (year % 4 == 0)
  {
    printf("%d is a leap year", year);
  }
  else if (year % 4 == 0 && year % 100 != 0)
  {
    printf("%d is a leap year", year);
  }
  else
  {
```
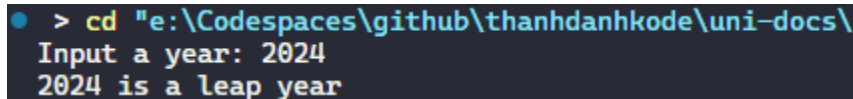
```c
    printf("%d is not a leap year", year);
  }

  return 0;
}
```

Result


```
 ● > cd "e:\Codespaces\github\thanhdanhkode\uni-docs\
   Input a year: 2024
   2024 is a leap year
```

Figure 2. Result of the program in Problem 2 – Leap year checking

Problem 3:

```c
#include <stdio.h>

/**
 *
 * This program check input is the triangle or not, and print what type of that triangle
 *
 */
int main()
{
  int x, y, z;

  printf("Input three sides of triangle: "); // Prompt user to input three sides of a
triangle
  scanf("%d %d %d", &x, &y, &z);

  if (x + y > z && x + z > y && y + z > x)
  {
    if (x == y && x == z)
    {
      printf("This is an equilateral triangle");
    }
    else if (x == y || x == z || y == z)
    {
      printf("This is an isosceles triangle");
    }
    else
    {
      printf("This is a scalene triangle");
    }
  }
  else
  {
    printf("This is not a triangle");
  }

  return 0;
}
```

Result

Figure 3. Result of the program in Problem 3 – Triangle types checking

Problem 4:

```c
#include <stdio.h>

/**
 *
 * This program calculate the diameter, circumference, area of a circle with custom
unit
 *
 */
int main()
{
  float radius;
  char unit[5];

  printf("Enter the unit: ");
  scanf("%s", unit);
  printf("Enter the radius of circle: ");
  scanf("%f", &radius);

  printf("Diameter of circle: %0.2f %s\n", 2 * radius, unit);
  printf("Circumference of circle: %0.2f %s\n", 2 * 3.14 * radius, unit);
  printf("Area of circle: %0.2f sq. %s\n", 3.14 * radius * radius, unit);

  return 0;
}
```

Result

Figure 4. Result of program in Problem 4 – Cacluate diameter, circumference and area of circle with radius and unit

Problem 5:

```c
#include <stdio.h>
```

```c
#include <math.h>

/**
 *
 * This program calculate the roots of the second order equation
 *
 */
int main()
{
  float a, b, c, x1, x2, delta = 0;

  printf("Enter the value of a, b, c: ");
  scanf("%f %f %f", &a, &b, &c);

  delta = pow(b, 2) - 4 * a * c;
  if (a == 0 && b == 0 && c == 0)
    printf("This is not a quadratic equation");
  else if (delta > 0)
  {
    x1 = (-b + sqrt(delta)) / (2 * a);
    x2 = (-b - sqrt(delta)) / (2 * a);
    printf("The root of equation are x1 = %0.2f, x2 = %0.2f\b", x1, x2);
  }
  else if (delta == 0)
    printf("Equation has double roots: %f", -b / (2 * a));
  else
  {
    printf("Equation has no real root");
  }

  return 0;
}
```
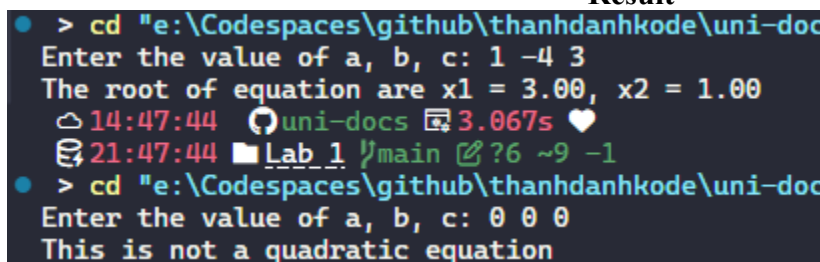
Result

Figure 5. Result of program in Problem 5 – Find the roots of an equation

Problem 6:

```c
#include <stdio.h>

/**
 *
 * This program calculate the total payment of Taxi service
 *
 */
int main()
{
```

```c
    float distance;
    float base_price = 10000;

    printf("Enter total distance in km: ");
    scanf("%f", &distance);

    if (distance <= 1)
      printf("The total cost that a passenger has to pay: %0.0f VND", base_price);
    else if (1 < distance <= 30)
      printf("The total cost that a passenger has to pay: %0.0f VND", base_price +
(distance - 1) * 5000);
    else
      printf("The total cost that a passenger has to pay: %0.0f VND", base_price + 30.00
* 5000 + (distance - 31) * 30000);

    return 0;
}
```
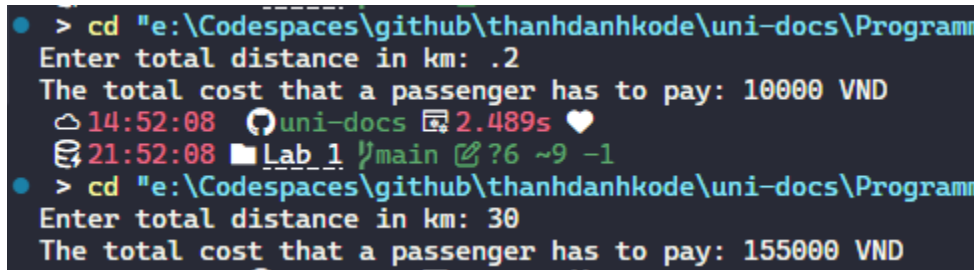
Result

Figure 6. Result of program in Problem 6 – Calculate the price that passenger has to pay

<u>Problem 7:</u>

```c
#include <stdio.h>

/**
 *
 * This program get student name, id, mark of literature, math, english and print out
their name, student id, their mark, total mark and average mark of 3 subjects above
 *
 */
int main()
{
    int id;
    float average_mark, literature_mark, mathematic_mark, english_mark;
    char name[100], student_id[20];

    printf("Input the Name of the student: ");
    scanf("%s", name);
    printf("Input the No of the student: ");
    scanf("%s", student_id);
    printf("Input the marks of Literature, Math and English: ");
    scanf("%f %f %f", &literature_mark, &mathematic_mark, &english_mark);

    average_mark = (literature_mark + mathematic_mark + english_mark) / 3;
```
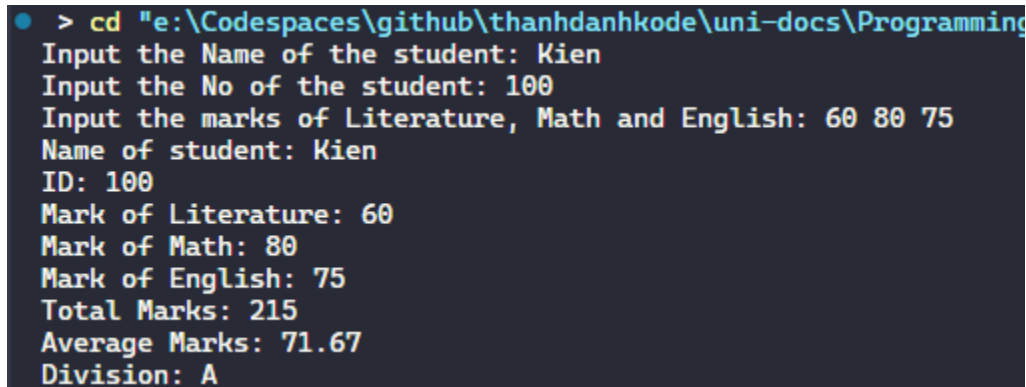
```c
    printf("Name of student: %s\n", name);
    printf("ID: %s\n", student_id);
    printf("Mark of Literature: %0.0f\n", literature_mark);
    printf("Mark of Math: %0.0f\n", mathematic_mark);
    printf("Mark of English: %0.0f\n", english_mark);
    printf("Total Marks: %0.0f\n", literature_mark + mathematic_mark + english_mark);
    printf("Average Marks: %0.2f\n", average_mark);

    if (average_mark >= 60)
      printf("Division: A");
    else if (48 <= average_mark < 60)
      printf("Division: B");
    else if (36 <= average_mark < 48)
      printf("Division: Pass");
    else if (average_mark < 36)
      printf("Division: Fail");
    else
      printf("error");

    return 0;
}
```

Result

Figure 7. Result of program in Problem 7 – Calcuate Student mark and divide into grade

Bonus 1:

```c
#include <stdio.h>

int main()
{
  int account_number;
  float begin_balance, total_charges, total_credits, credit_limit, new_balance;

  while (1)
  {
    printf("\nEnter account number (-1 to end): ");
    scanf("%d", &account_number);
    if (account_number == -1)
      break;
    printf("Enter beginning balance: ");
    scanf("%f", &begin_balance);
```
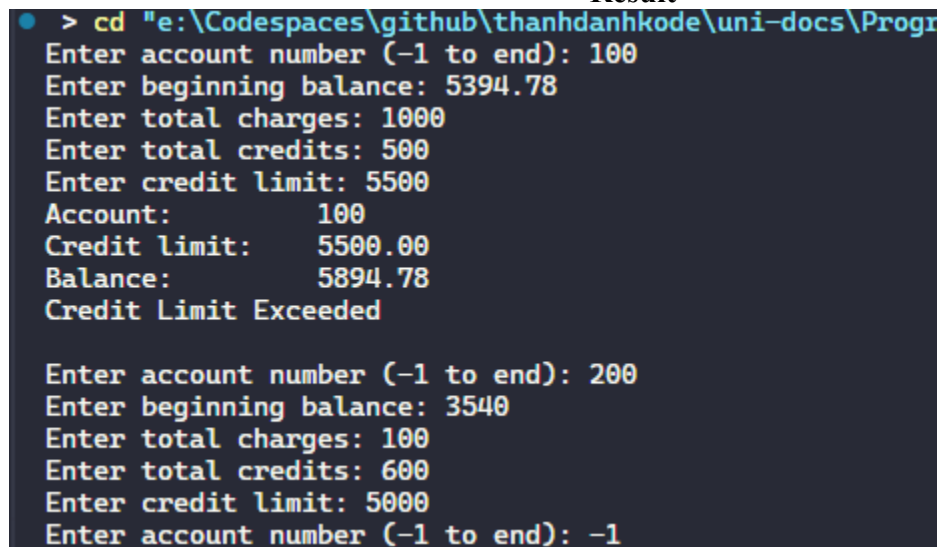
```
    printf("Enter total charges: ");
    scanf("%f", &total_charges);
    printf("Enter total credits: ");
    scanf("%f", &total_credits);
    printf("Enter credit limit: ");
    scanf("%f", &credit_limit);

    new_balance = begin_balance + total_charges - total_credits;
    if (credit_limit < new_balance)
    {
      printf("Account: \t %d\n", account_number);
      printf("Credit limit: \t %.2f\n", credit_limit);
      printf("Balance: \t %.2f\n", new_balance);
      printf("Credit Limit Exceeded\n");
    }
  }

  return 0;
}
```

Result

Figure Bonus 1. Result of program in Bonus 1 – Check credit limit is exceed of a user

Bonus 2:

```
#include <stdio.h>

int main()
{
  int x, tmp;
  int reminder, reversedNum = 0;

  printf("Enter 5-digits integers: ");
  scanf("%d", &x);
  tmp = x;

  while (tmp != 0)
```
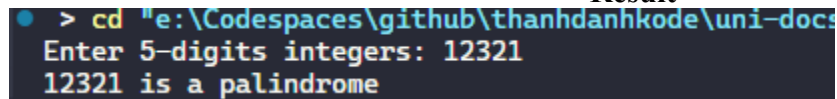
```
    {
      reminder = tmp % 10;
      reversedNum = reversedNum * 10 + reminder;
      tmp /= 10;
    }

    if (reversedNum == x)
      printf("%d is a palindrome", x);

    return 0;
}
```

Result

Figure Bonus 2. Result of program in Bonus 2 – Check 5-digits is a palidrome

Bonus 3:

```
#include <stdio.h>
/**
 *
 * This program covert a Binary number to Decimal number using iterations
 *
 */
int main()
{
  int x, base = 1, decimal = 0;

  printf("Enter binary value: ");
  scanf("%d", &x);

  while (x != 0)
  {
    int reminder = x % 10;
    x /= 10;
    decimal += reminder * base;
    base *= 2;
  }

  printf("Decimal: %d", decimal);

  return 0;
}
```
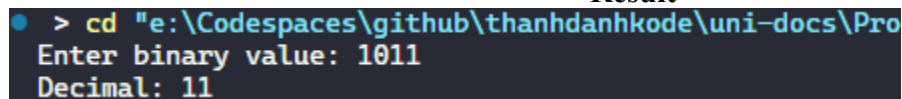
Result

Figure Bonus 3. Result of program in Bonus 3 – Conver binary into decimal

Conclusion

In this lab, I learned how to write and test C programs to solve different computational problems, such as calculating equations, determining conditions, and processing user input. I practiced using conditional statements, loops, and arithmetic operations.

At first, I had some difficulties understanding how to apply logical conditions correctly and handle user input properly in the C language. I overcame these problems by reviewing lecture notes, looking up syntax examples, and testing my code step by step to find and fix errors.

The most important part of this lab was learning how to think logically and structure programs clearly using pseudocode before writing the actual code. This helped me improve my problem-solving skills and confidence in programming with C.

THE END