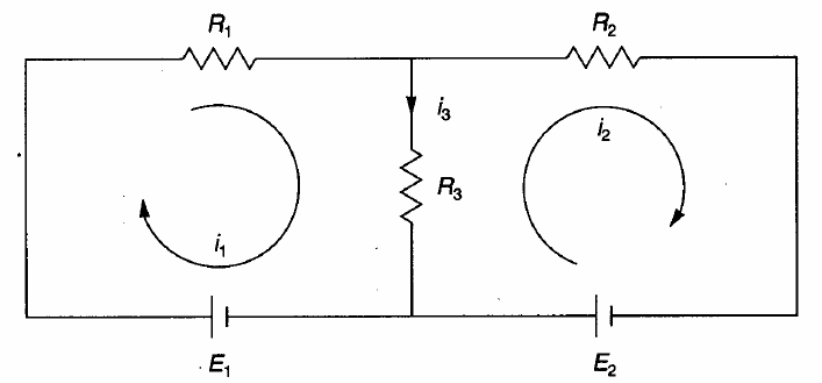
BIL105E
SELF STUDY EXERCISES

PART 1 : CALCULATIONS AND SELECTIONS

QUESTION 1) Electric Currents

The following is an electrical circuit. R_1, R_2, R_3 are resistors; E_1, E_2 are voltage sources; I_1, I_2, I_3 are currents.



The currents I_1, I_2, and I_3 can be determined using the following equations:

$$I_1 = \frac{E_2 R_3 + E_1 (R_1 + R_3)}{(R_1 + R_3)(R_2 + R_3) - (R_3)^2}$$

$$I_2 = \frac{E_1 R_3 + E_2 (R_1 + R_3)}{(R_1 + R_3)(R_2 + R_3) - (R_3)^2}$$

$$I_3 = I_1 - I_2$$

Draw a Flow Chart and write a C program to compute and display the currents. The values of R_1, R_2, R_3, E_1, E_2 should be entered from the keyboard.

QUESTION 2) Determining Even or Odd

Draw a flowchart and write a C program which reads an integer number from the user, then displays a message whether the given number is even or odd.

Rule: If a number can be divided by 2 without any remainder, then it is considered as even, otherwise odd.

QUESTION 3) Determining the Smallest

Draw a flowchart and write a C program which reads three numbers from the user, then displays the smallest number.

QUESTION 4) Quadrants

In coordinate system, the quadrant that a line drawn from the origin resides in is determined by the angle that the line makes with the positive X axis as follows:

Angle from the Positive X Axis	Quadrant
Between 0 and 90 degrees	I
Between 90 and 180 degrees	II
Between 180 and 270 degrees	III
Between 270 and 360 degrees	IV

Draw a flowchart and write a C program that accepts the angle of the line as user input and determines and displays the quadrant.

Note: If the angle is exactly 0, 90, 180, or 270 degrees the corresponding line does not reside in any quadrant but lies on an axis.

PART 2 : LOOPS

QUESTION 5) Generating a Table

Draw a flowchart and write a C program which reads an integer number N from the user, then displays the following table.

```
 X       10*X       100*X
====    ======    =======
 1       10         100
 2       20         200
 3       30         300
 4       40         400
. . .    . . .      . . .
. . .    . . .      . . .
. . .    . . .      . . .
 N       . . .      . . .
```

QUESTION 6) Temperature Table

Draw a flowchart and write a C program to convert Celsius degrees (from 0 to N, in steps of 5 degrees) to Fahrenheit.

Use the following formula:

Fahrenheit = (9.0 / 5.0) * Celsius + 32.0

The program should request the N value for the Celsius value.

Your output should be like the the following:

DEGREES CELSIUS	DEGREES FAHRENHEIT
0	32.00
5	41.00
10	50.00
15	59.00

20	68.00
25	77.00
30	86.00
. . .	. . .
N	. . .

QUESTION 7) Distance Calculation

S is distance, **t** is time, and **a** is acceleration. The following is the distance formula of a moving vehicle.

$$S = \frac{1}{2}at^2$$

Draw a flowchart and write a C program that computes the **S** distances starting from t=0, up to t=N, in steps of 1.

The values of **a** and **upper N** will be entered by user.

QUESTION 8) Displaying Shape

Draw a flowchart and write a C program to print a shape as the following.

```
* * * * * * * *
 * * * * * * *
  * * * * * *
   * * * * *
    * * * *
     * * *
      * *
       *
```

The number of lines (**N)** value will be entered by user.

QUESTION 9) Factorial

Draw a flowchart and write a C program to calculate the factorial of a positive integer number.

The N value will be entered by user.

If the user enters a negative number, then a warning message should be displayed.

N ! = 1 * 2 * 3 * 4 * …….. * N

PART 3 : SERIALS

QUESTION 10) Calculating e^x

e^x can be computed by using the following formula.

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \ldots\ldots\ldots + \frac{x^n}{n!}$$

Draw a flowchart and write a C program to calculate and display e^x value.

The **X** and **N** values will be entered by user.

QUESTION 11) Sum of Series

T is the sum of the following series.

$$T = \frac{1}{1*3^1} + \frac{1}{2*3^2} + \ldots \frac{1}{N*(3^N)}$$

$$= \sum_{i=1}^{N} \frac{1}{i*(3^i)}$$

Draw a flowchart and write a C program to calculate and display the T value.

The **N** value will be entered by user from keyboard.

QUESTION 12) Sum of Series

T is the sum of the following series.

$$T = \sum_{k=0}^{N} \frac{(-1)^k X^{2k}}{(2k)!}$$

Draw a flowchart and write a C program to calculate and display the T value. The **X** and **N** values will be entered by user.

QUESTION 13) Sum of Sum of Series

S is the sum of the sum as defined below.

$$S = \sum_{i=1}^{N} \sum_{j=1}^{M} (i + j)^2$$

Draw a flowchart and write a C program to calculate and display the S value. The **N** and **M** values will be entered by user.

QUESTION 14) Difference of Sums

D is the difference as defined below.

$$D = \left(\sum_{i=1}^{N} i \right)^2 - \left(\sum_{i=1}^{N} i^2 \right)$$

Draw a flowchart and write a C program to calculate and display the D value.

The **N** value will be entered by user.

PART 4 : AVERAGES

QUESTION 15) Mean Values

Draw a flowchart and write a C program to calculate Arithmetic, Geometric, Harmonic, and Quadratic mean values of a set of N real numbers (X_1, X_2, . . . X_N)

The X values will be entered by the user.

$$m_A = \frac{\sum_{i=1}^{N} X_i}{N} \qquad m_G = \sqrt[N]{\prod_{i=1}^{N} X_i}$$

$$m_H = \frac{N}{\sum_{i=1}^{N} \frac{1}{X_i}} \qquad m_Q = \sqrt{\frac{\sum_{i=1}^{N} {X_i}^2}{N}}$$

QUESTION 16) Averages of Arrays

Assume that, 4 experiments are performed, each experiment consisting of 6 test results.

The results for each experiment are given below.

```
1st experiment results:  3   8  1   6  1   4
2nd experiment results:  6   5  9   2  10  2
3rd experiment results:  7   3  20  1  4   4
4th experiment results:  1   4  2   9  3   6
```

Draw a Flow Chart and write a C program using a nested loop to

compute and display the average of the test results for each experiment. (All data values should be defined in arrays, so keyboard input is not required.)

QUESTION 17) Using Structs

a. Declare a single **structure** template suitable for a car record of the type illustrated below.

Car No.	Miles Driven	Gallons Used
54	250	19
62	540	37
71	127	6
85	830	74
97	235	16

b. Using the template you declared above, write a C program that interactively accepts the above data into an array of five structures.

Once the data have been entered, the program should create a report listing each car number and the miles per galon achieved by the car.

At the end of the report include the average miles per galon achieved by the complete fleet of cars.

PART 5 : NUMBER THEORY

QUESTION 18) Checking Prime Number

A prime number is a positive integer that is greater than 1 and has no positive integer divisors other than 1 and itself (without remainder). The first ten prime numbers are: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, . . .

(Note: 1 is not a prime number. The smallest prime number is 2)

Draw a flowchart and write a C program which displays a message about whether the given number is prime or not.

The number will be entered by the user. For example, when the user enters 18 then program should display “It is not a prime”. When the user enters 7 then program should display “It is a prime”.

QUESTION 19) Displaying All Prime Numbers

Draw a flowchart and write a C program which displays all prime numbers up to the **L**.

Example Primes: 2 , 3 , 5 , 7 , 11, 13, 17, 19, 23, 29, 31, 37, . . . , L

The value of the Upper Limit (**L**) will be entered by the user.

QUESTION 20) Perfect Numbers

A number is called Perfect if all its integer divisors (including 1) sum to itself.

Examples:

6 is a perfect number, because 6 = 1+2+3

28 is a perfect number, because 28 = 14+7+4+2+1

Draw a flowchart and write a C program that finds and displays all perfect numbers between 1 and 1000.

QUESTION 21) Friendly Numbers

Two positive integers are friendly if each one is equal to the sum of the divisors (including one and excluding the number itself) of each other.

For example, 220 and 284 are friendly.

Divisors of 220 = 1 , 2 , 4 , 5 , 10 , 11 , 20 , 22 , 44 , 55 , 110
Sum of 220's divisors = 1+2+4+5+10+11+20+22+44+55+110 = 284

Divisors of 284 = 1 , 2 , 4 , 71
Sum of 284's divisors = 1+2+4+71+142 = 220

Write a C program to find and display all pairs of friendly numbers between 1 and 10000.

(WARNING: PROGRAM MAY BE TOO SLOW)

QUESTION 22) Armstrong Numbers

Write a program to print out all Armstrong numbers between 100 and 999.

If sum of cubes of each digit of the number is equal to the number itself, then the number is called an Armstrong number.

For example, 153 = 13 + 53 + 33
= 1 + 125 + 27
= 153

PART 6 : DIGITS

QUESTION 23) Bills and Coins

Write a C program to calculate and display the exact bills and coins for a purchase. The program should use as bigger bills and coins as possible.

The input is the purchase value using float notation in TL and Kuruş.

For example 2410,80 means 2410 TL and 80 Krş

The output is the number of Turkish coins and bills.

Available bills are 5 TL, 10 TL, 20 TL, 50 TL, 100 TL, 200 TL, 500 TL.
Available coins are 1 Krş, 5 Krş, 10 Krş, 25 Krş, 50 Krş, and 1 TL.

Example output for 2410,80

4 tane 500 TL = 2000
2 tane 200 TL = 400
1 tane 10 TL = 10

1 tane 50 Krş = 50
1 tane 25 Krş = 25
1 tane 5 Krş = 5

QUESTION 24) Digits of a Number

Draw a flowchart and write a C program to display the digits of a positive integer nember. The number will be entered by user. In your output, seperate the digits with the “-“ character.

Example:

When the user enters 2750

then your output should be 2-7-5-0

PART 7 : STRINGS

QUESTION 25) Strings

a. Write a C function to count the total number of characters, including blanks, contained in a string.
Do not include the end-of-string marker in the count.

b. Include the function you wrote above in a complete working program.

c. Write a C program that accepts a string of characters from a terminal and displays the string one word per line.

d. Write a C function that reverses the characters in a string.
(Hint: This can be considered as a string copy starting from the back end of the first string.)

e. Write a C function called del_char ( ) that can be used to delete characters from a string.
The function should take three arguments: the string name, the number of characters to delete, and the starting position in the string where characters should be deleted.

For example, the function call del_char (mystring, 13, 5) , when applied to the string *“all enthusiastic people”*, should result in the string *“all people”*.

f. Write a C function call add_char ( ) to insert one string of characters into another string.
The function should take three arguments: the string to be inserted, the original string, and the position in the original string where the insertion should begin.
For example, the call add_char ("for all", message, 6) should insert the characters *“for all”* in message starting at message at position 6 .

g. Write a C program that counts the number of words in a string. Assume that the seperator characters are :
(blank space), (,) (;) (.) (?) (-)