

Practical Workbook

CT-175

**PROGRAMMING
FUNDAMENTALS**



Name: Hamna Ali Khan

Year: 2024

Batch: 2028

Roll No: CT-157

Department: BCIT

Dept. of Computer Science & Information Technology
NED University of Engineering & Technology

EXERCISE Q# 01

Write pseudo-code of a program for finding the maximum out of three numbers.

BEGIN

VARIABLES a, b, c

INPUT a

INPUT b

INPUT c

IF $a > b$ and $b > c$, then

 OUTPUT "a is greater"

ELSEIF $b > a$ and $b > c$, then

 OUTPUT "b is greater"

ELSE

 OUTPUT "c is greater"

END

EXERCISE Q# 02

Write pseudo-code and draw flow chart. Ask a user to enter exam scores for five different courses and determine whether the student is passing or failing the course. Calculate the average score, the number of failed courses, and the number of passed courses To confirm your solution, trace through the designed flow chart and pseudo-code by using the following test case: 88, 65, 45, 23, 77

BEGIN

VARIABLE passed_courses= 0, failed_courses=0, total_scores= 0, number_of_courses= 5,
passing_marks= 50, avg_scores

INPUT number of courses

FOR i=1 to number of courses

INPUT score

IF score >= 50, then

OUTPUT "course passed"

ADD 1 to passed_courses

ELSE

OUTPUT "course failed"

ADD 1 to failed_courses

ENDIF

ADD score to total_scores

ENDFOR

avg_score= total_scores/number_of_courses

OUTPUT "Passed courses:", passed_courses

OUTPUT "Failed courses:", failed_courses

OUTPUT "Average scores:", avg_scores

END

OUTPUT OF PSEUDO-CODE:

DRY RUN

Input number of courses= 5

Loop iterations:

Iteration 1: (i=1)

Input: 88

Condition: 88 >= 50 → TRUE

Action: Course passed

Update: Passed course= 1, total scores= total scores + 88

Iteration 2: (i=2)

Input: 65

Condition: 65 >= 50 → TRUE

Action: Course passed

Update: Passed course= 2, total scores= total scores + 65=153

Iteration 3: (i=3)

Input: 45

Condition: 45 >= 50 → FALSE

Action: Course failed

Update: Failed course= 1, total scores= total scores + 45=198

Iteration 4: (i=4)

Input: 23

Condition: $23 \geq 50 \rightarrow \text{FALSE}$

Action: Course failed

Update: Failed course= 2, total scores= total scores + 23=221

Iteration 5: (i=5)

Input: 77

Condition: $77 \geq 50 \rightarrow \text{TRUE}$

Action: Course passed

Update: Passed course= 3, total scores= total scores + 77=298

After loop process:

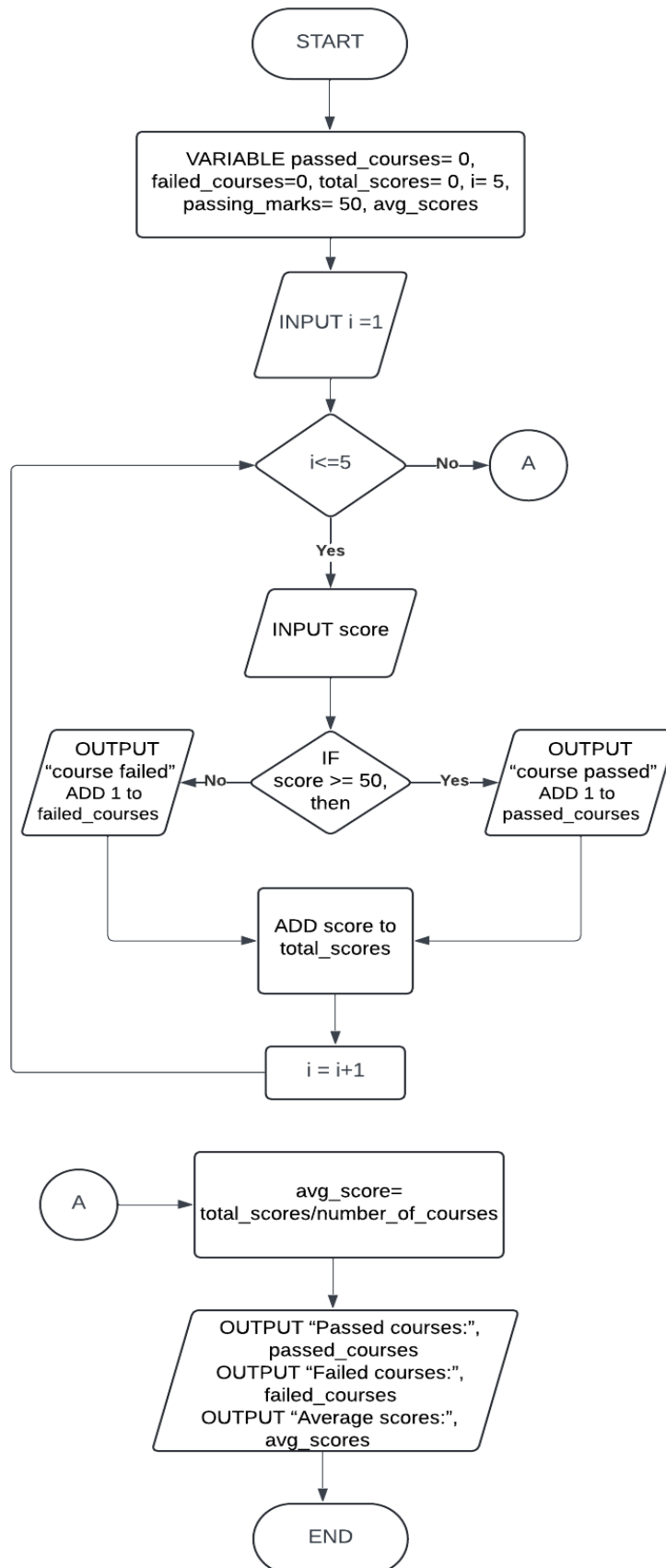
Average scores: total scores/number of courses= $298/5 = 59.6$

Final Output:

“Passed courses: 3”

“Failed courses: 2”

“Average scores: 59.6”



EXERCISE Q# 03

Ask a user to enter a number and then display the factorial of the entered number.

BEGIN

VARIABLE a, number, factorial=1

INPUT number

FOR a=1 to number

 factorial =factorial * a

ENDFOR

OUTPUT “Factorial of”, number, “is”, factorial

END

Source code:

```
#include <stdio.h>
int main() {
    int number;
    printf("Enter a number: ");
    scanf("%d", &number);

    int factorial = 1;
    for (int a = 1; a <= number; a++) {
        factorial *= a;
    }

    printf("Factorial of %d is %d\n", number, factorial);

    return 0;
}
```

OUTPUT:

```
Enter a number: 5
Factorial of 5 is 120
```


EXERCISE Q# 04

A board meeting that Joe Roberts has been given at work is to order special paper for a report for one of the jobs Joe comes in reams of 500 sheets. He always makes five more copies than the number of people that will be there. Joe wants to know how many reams of paper he needs for a meeting. He can order only whole reams. Test your solution with the following data: The number of pages will not equal an exact number of reams, partial reams. Assume the required number of pages is 140 pages long. There will be 25 people at the meeting.

BEGIN

VARIABLES reportlength= 140, attendees= 25, reamsneeded, reamsofsheets= 500,extracopies=5, totcopies

totcopies= attendees+extracopies

Sheetsneeded= reportlength*totcopies

Reamsneeded= ceil(sheetsneeded/reamsofsheets)

OUTPUT "reamsneeded"

END

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

```
#include <math.h>
```

```
int main() {
```

```
    int reportLength = 140;
```

```
    int attendees = 25;
```

```
    int copies = attendees + 5;
```

```
    int sheetsNeeded = reportLength * copies;
```

```
    int reamsNeeded = ceil((double)sheetsNeeded / 500);
```

```
    printf("Reams of paper needed: %d\n", reamsNeeded);
```

```
    return 0;
```

DRY RUN:

INPUT

Reportlength= 140

Attendees= 25

Reamsofsheets= 500

Extracopies= 5

PROCESS

Totcopies= $25 + 5 = 30$

Sheetsneeded= $140 * 30 = 4200$

Reamsneeded = $\text{ceil}(4200/500) = \text{ceil}(8.4) = 9$

OUTPUT

The number of reams that Joe needs to order = 9 reams

EXERCISE Q# 05

Joe would like to build several bookcases that have different heights and widths. All will be 12 inches in depth. The bookcases will have three shelves, in addition to the bottom and the top. Write a solution to print the number of feet of 12-inch-wide boards that Joe will need to complete a bookcase, given the height and width.

BEGIN

VARIABLES height, width, widthinfeet, shelves, totalmaterial

INPUT "Enter the height of the bookcase in inches:"

OUTPUT "height"

INPUT "Enter the width of the bookcase in inches:"

OUTPUT "width"

INPUT "Enter the number of shelves:"

OUTPUT "shelves"

widthinfeet = width / 12

totalmaterial = (shelves * widthinfeet) + (2 * height/12) + (2 * widthinfeet)

OUTPUT "Total feet of 12-inch-wide boards needed: ", totalmaterial

END

DRY RUN:**Suppose:**

height = 84 inches

width = 48 inches

Convert width to feet:

widthinfeet = $48 / 12 = 4$ feet

Calculate total material:

Shelves: $3 * \text{widthinfeet} = 3 * 4 = 12$ feet (for the 3 shelves)

Sides: $2 * (\text{height} / 12) = 2 * (84 / 12) = 2 * 7 = 14$ feet (for the two sides)

Top and Bottom: $2 * \text{widthinfeet} = 2 * 4 = 8$ feet (for the top and bottom)

Total Material:

totalmaterial = 12 (shelves) + 14 (sides) + 8 (top/bottom) = 34 feet

Output:

"Total feet of 12-inch-wide boards needed: 34 feet"

Summary of Values:

Shelves: 12 feet

Sides: 14 feet

Top and Bottom: 8 feet

Total: 34 feet