



Algorithms and Flowcharts

Section 4

Created by: Rehab Mohamed

Task

أوجد المخطط الاتسيلي (Flowchart) وكود الثفرة (Pseudo code) لخوارزم يقرأ أربعة أرقام (A, B, C, and D) ثم يقوم بإيجاد مجموع هذه الأرقام (Sum). إذا كان المجموع رقم زوجي فإن قيمة المتغير (Var) تساوي الجذر التربيعي للمجموع وإذا كان المجموع فردي فإن قيمة المتغير (Var) تساوي مربع المجموع. الخوارزم يقوم بطباعة المجموع (Sum) وقيمة المتغير (Var).



Pseudo code

Variable A, B, C, D, Sum : integer

Variable Var: real

Begin

Read (A, B, C, D)

Sum = A + B + C + D

If (Sum Mod 2 == 0) then

 Var = Sum ** 0.5

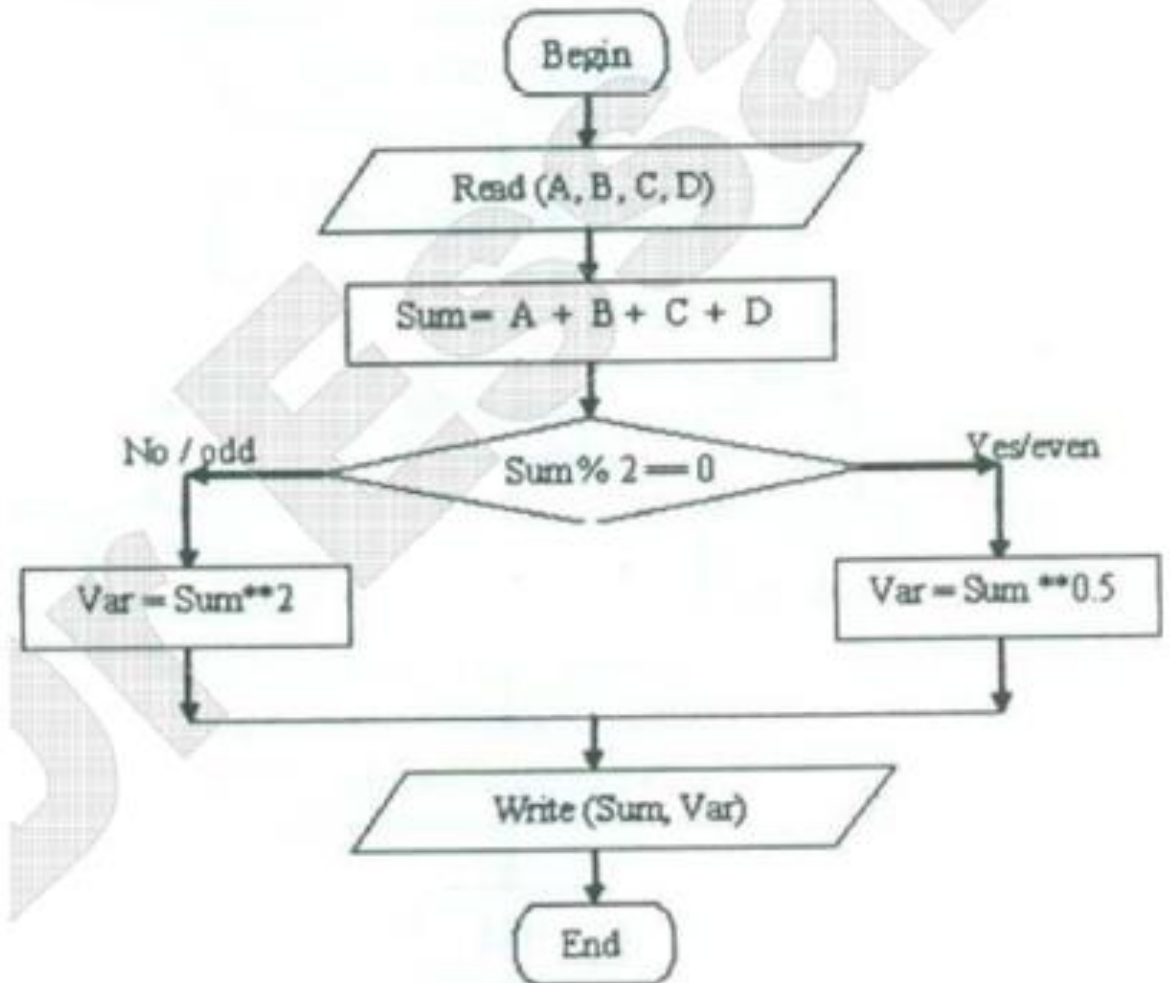
Else

 Var = Sum ** 2

End if

Write (Sum, Var)

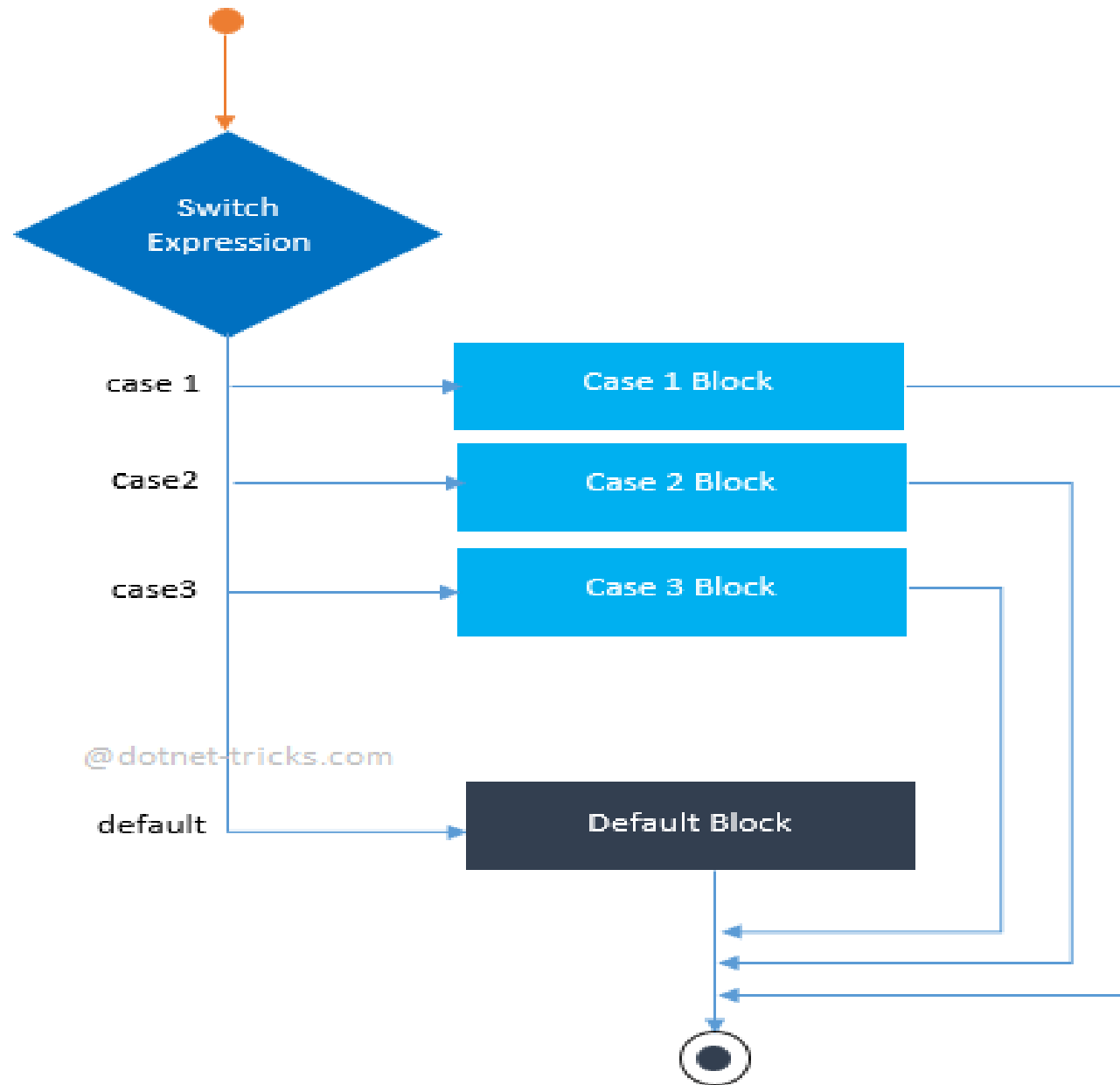
End



Switch Statement

Switch statement acts as a substitute for a long **if-else-if** that is used to test a list of cases. A switch statement contains one or more case labels which are tested against the switch expression. When the expression match to a case then the associated statements with that case would be executed.

```
Switch (expression)
{
  case value1:
    //Statements
    break;
  case value 2:
    //Statements
    break;
  case value 3:
    //Statements
  case value n:
    //Statements
    break;
  Default:
    //Statements
}
```



Switch Statement

Write algorithm which read the numbers from 1 to 7 and display their correspondence day of week.

Step 1: Start

Step2: read integer number say
(day)

Step3: Switch (day)

{

Case 1 :

Print "Saturday"

break;

Case 2 :

Print "Sunday"

break;

Case 3 :

Print "Monday"

break;

Case 4 :

Print "Tuesday"

break;

Case 5 :

Print "Wednesday"

break;

Case 6 :

Print "Thursday"

break;

Case 7 :

Print "Friday"

break;

Default :

Print "Invalid day"

break;

}

Step4: stop

Algorithm to calculate the area of A rectangle or circle or triangle by taking the user's choice.

Step 1: Start

Step 2: Initialize variables

Step 3: Take input for choice and then for area variables from the user

Step 4: Case 1: $ac = 3.14 * r * r$

break;

Case 2: $ar = \text{length} * \text{width}$

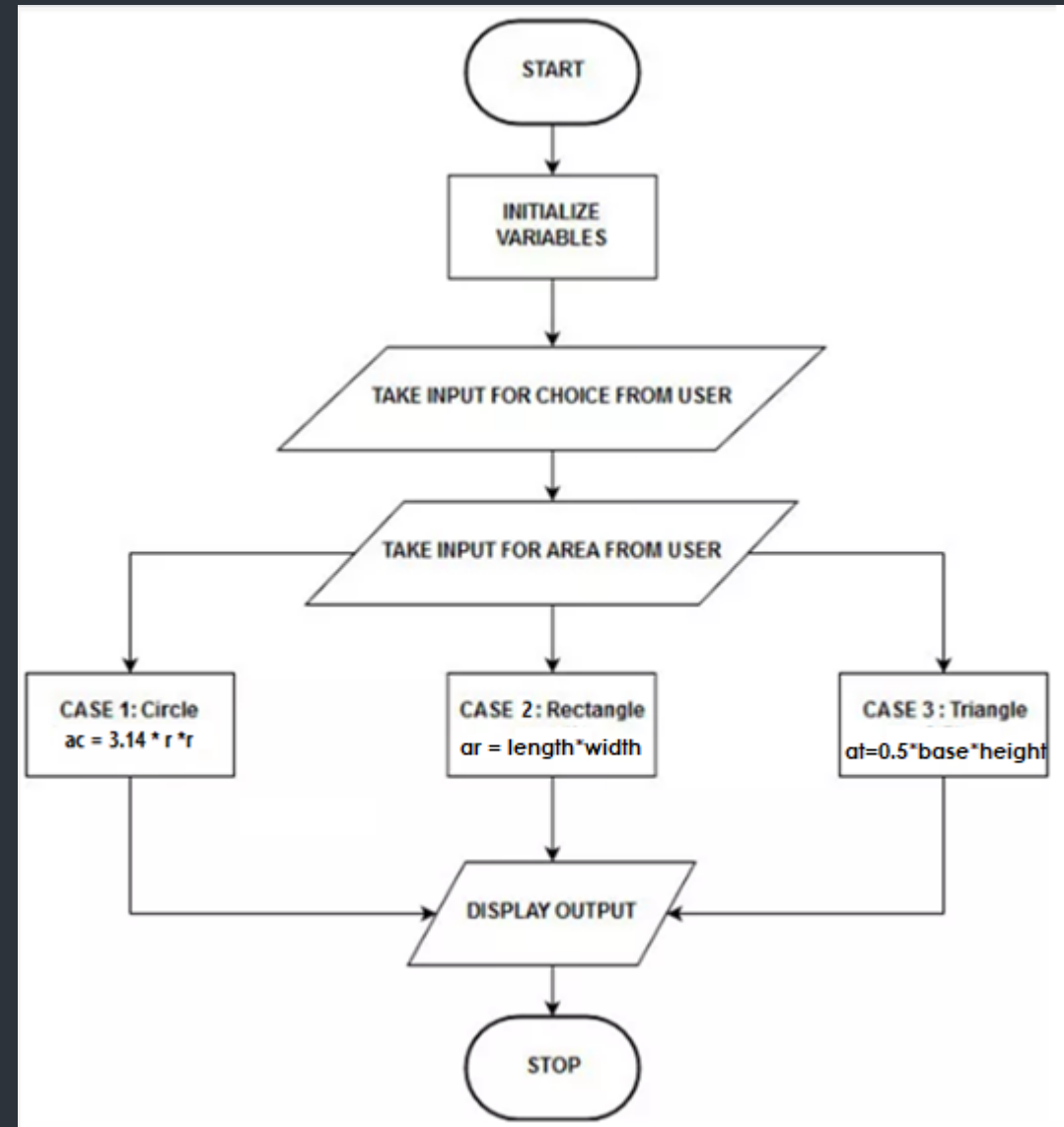
break;

Case 3: $at = 0.5 * \text{base} * \text{height}$

break;

Step 5: Display output according to case

Step 6: Stop



Perform arithmetic operations using switch statement

Step 1: Start

Step 2: Input character variable say (operator)

Step 3: Input two numbers say (num1 , num2)

Step 4: switch (operator)

{

case '+':

result = num1 + num2;

break;

case '-':

result = num1 - num2;

break;

case '*':

result = num1 * num2;

break;

case '/':

result = num1 / num2;

break;

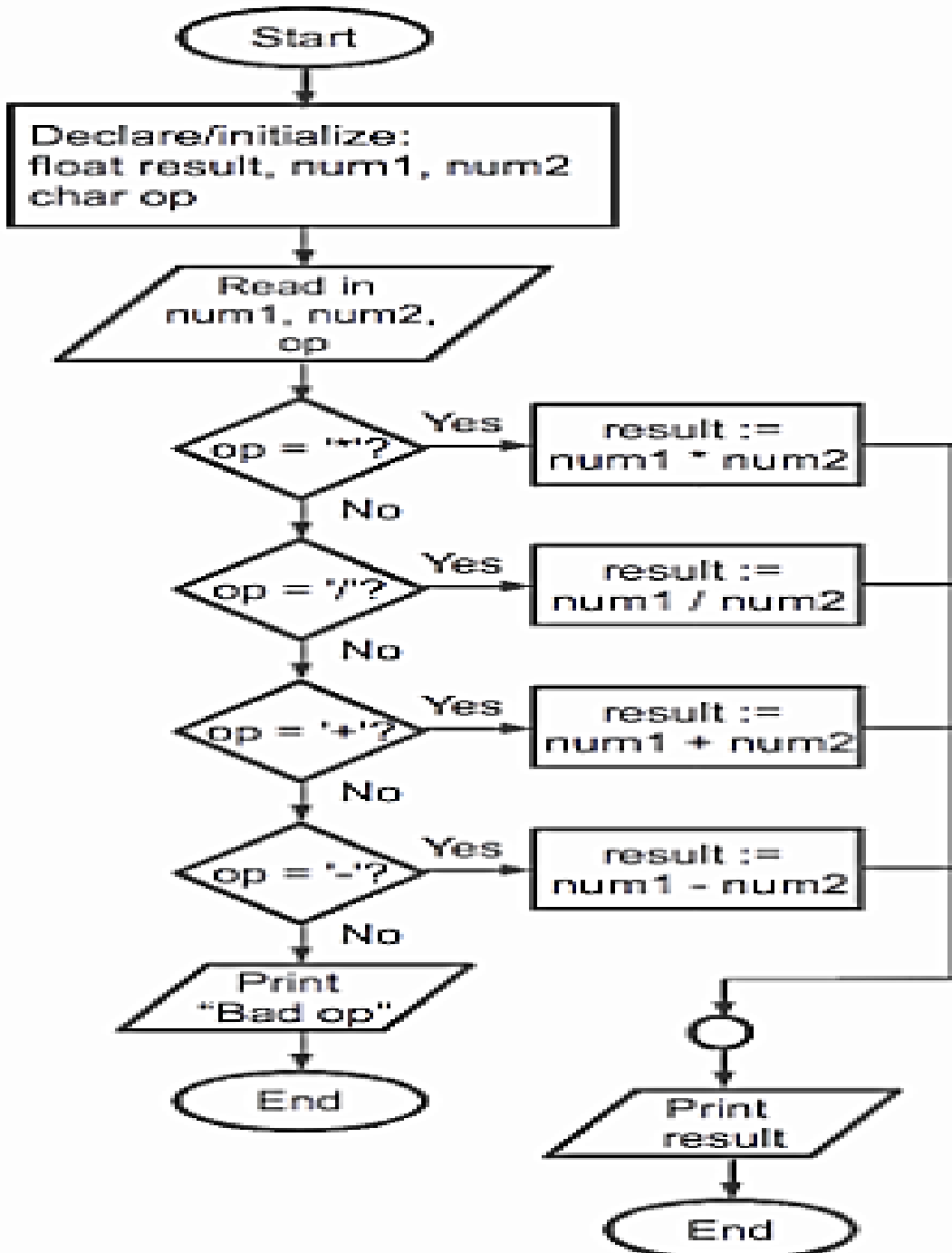
default:

print (Unknown Operator)

break;

Step 5: Display result

Step 6: Stop



Write algorithm that read a word English vocabulary from user as string and print its definition, at case of word not found print "The vocabulary is not found in the database".

Note: solve using if statement.

Word vocabulary	Definition
Pillow	Is soft or hard thing that you put your head on it.
Pillowcase	Is a cover for the pillow
Dresser	Is a piece of furniture that you put your clothes into

Algorithm

Step 1: Start

Step 2: Initialize variable (word) as string

Step 2: Read word

Step 3: if (word == "Pillow")

Print (Is soft or hard thing that you
put your head on it)

else if (word == "Pillowcase")

Print (Is a cover for the pillow)

else if (word == "Dresser")

Print (Is a piece of furniture that you

put your clothes into)


else

Print (The vocabulary is not found in
the database)

Step 4: stop



Problem solving with loops



➤ A loop statement allows us to execute a statement or group of statements multiple times.

➤ **Types of Loop Structures**

1. Automatic Counter Looping (ACL)

➤ For-Counter Loop

2. Conditioned Looping (CL)

➤ While Loop

➤ Do-While Loop

➤ Repeat-Until Loop

For loop

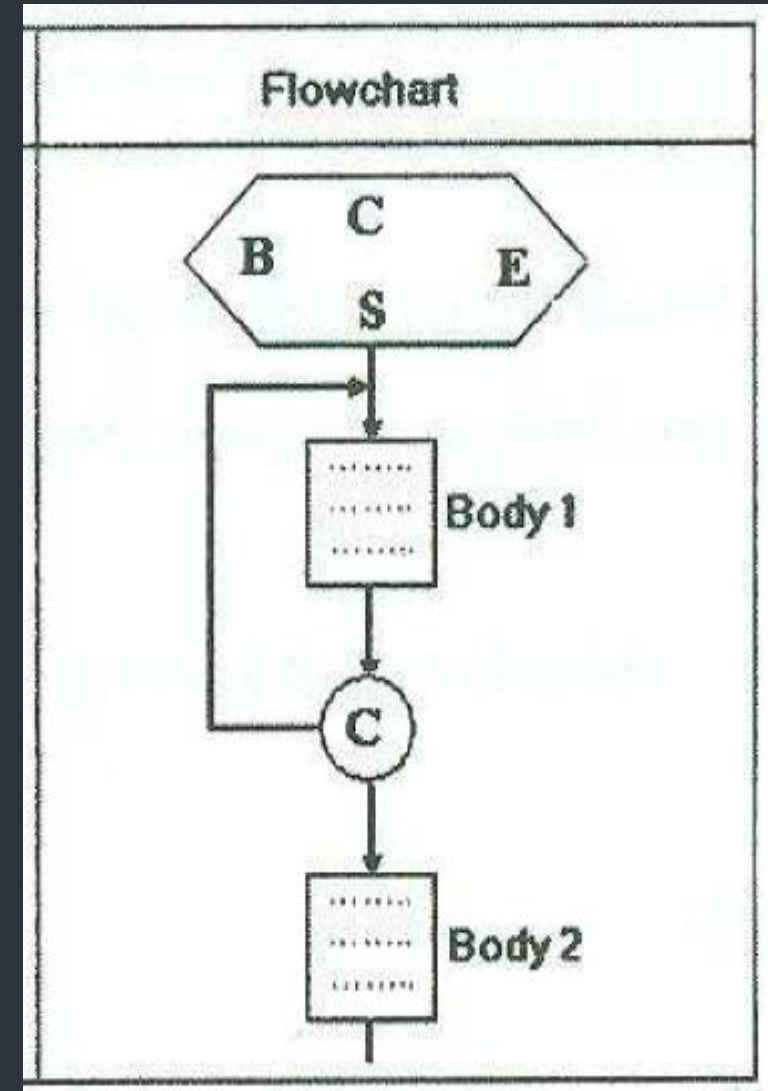
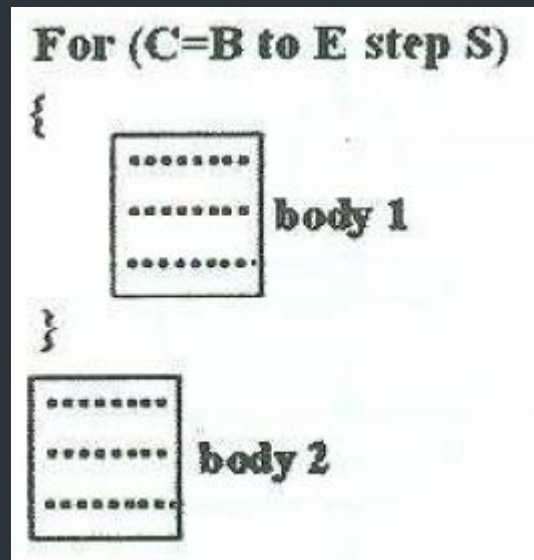
➤ A FOR loop is a loop that repeats a specified number of times. The loop uses a counter to tell it how many times to run the same sequence of activities.

➤ Syntax

Loop : Counter = Begin to End step S

body

Loop-End: Counter



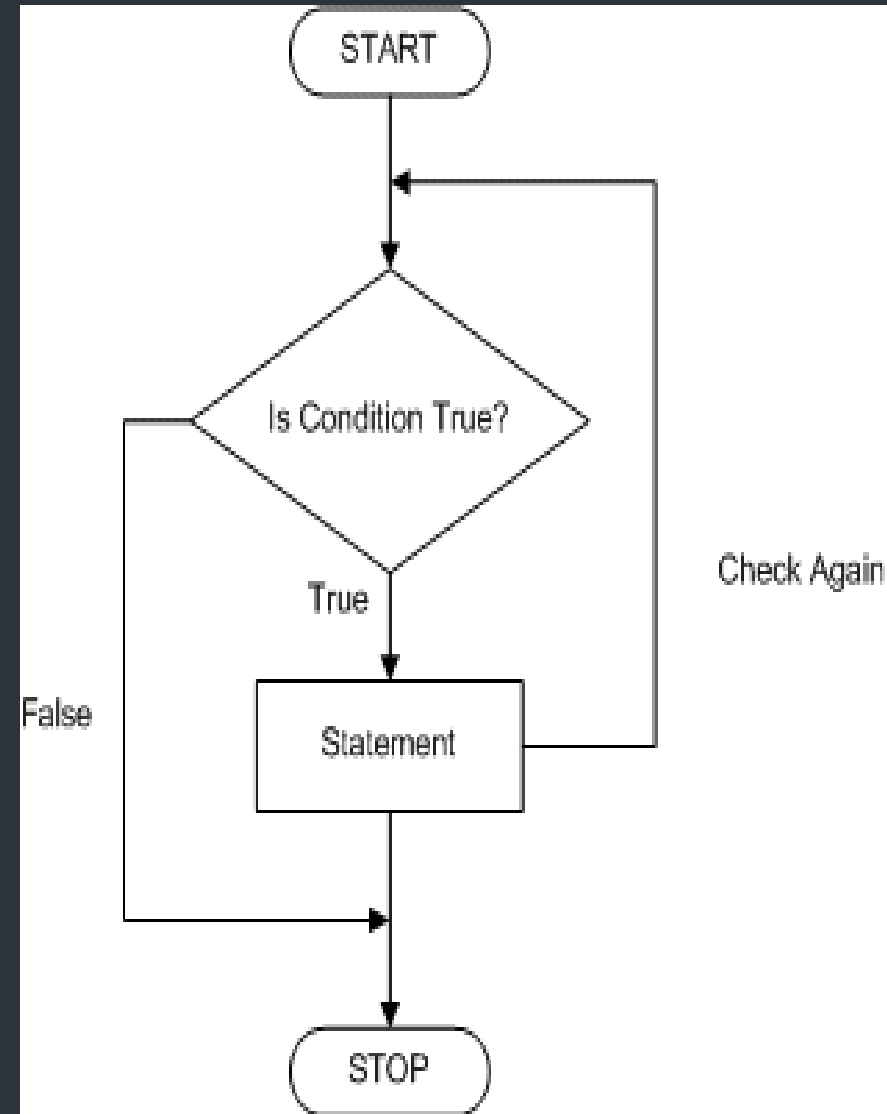
While Loop

This type of loop tells the computer that while the condition is TRUE, repeat all instructions between the WHILE and the WHILE END.

Syntax:

```
WHILE condition  
  Instructions  
END-WHILE
```

The statement is executed as long as the condition is True. The loop terminates when the condition is False.



Repeat/Until loop

- ▶ This type of loop tells the computer to repeat the set of instructions between the REPEAT and until, until a condition is TRUE.

- ▶ Syntax

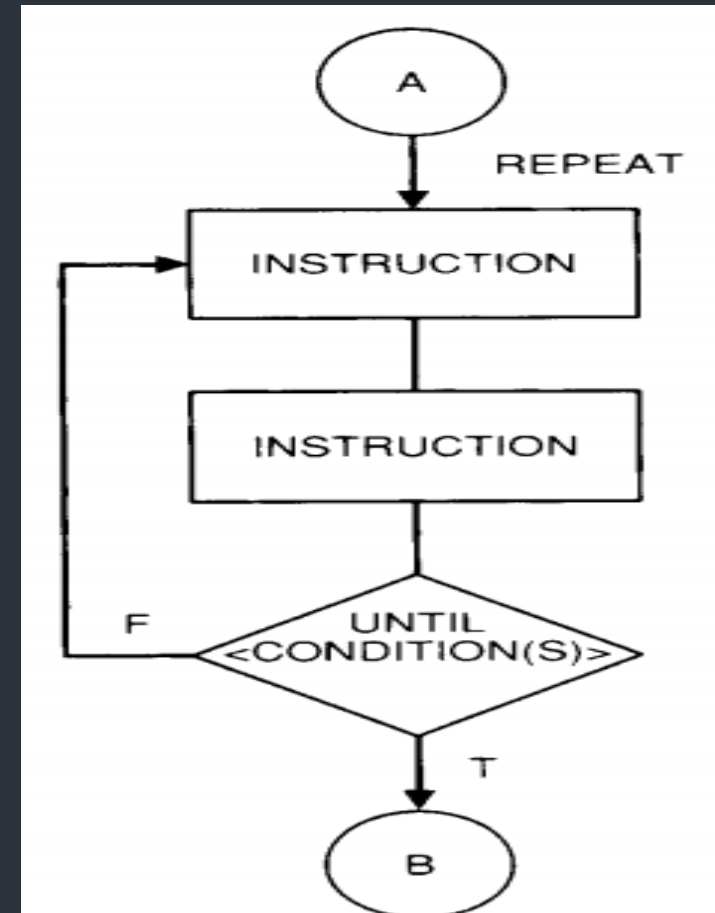
Repeat

Instruction

Instruction

...

Until <condition>



Do-While Loop

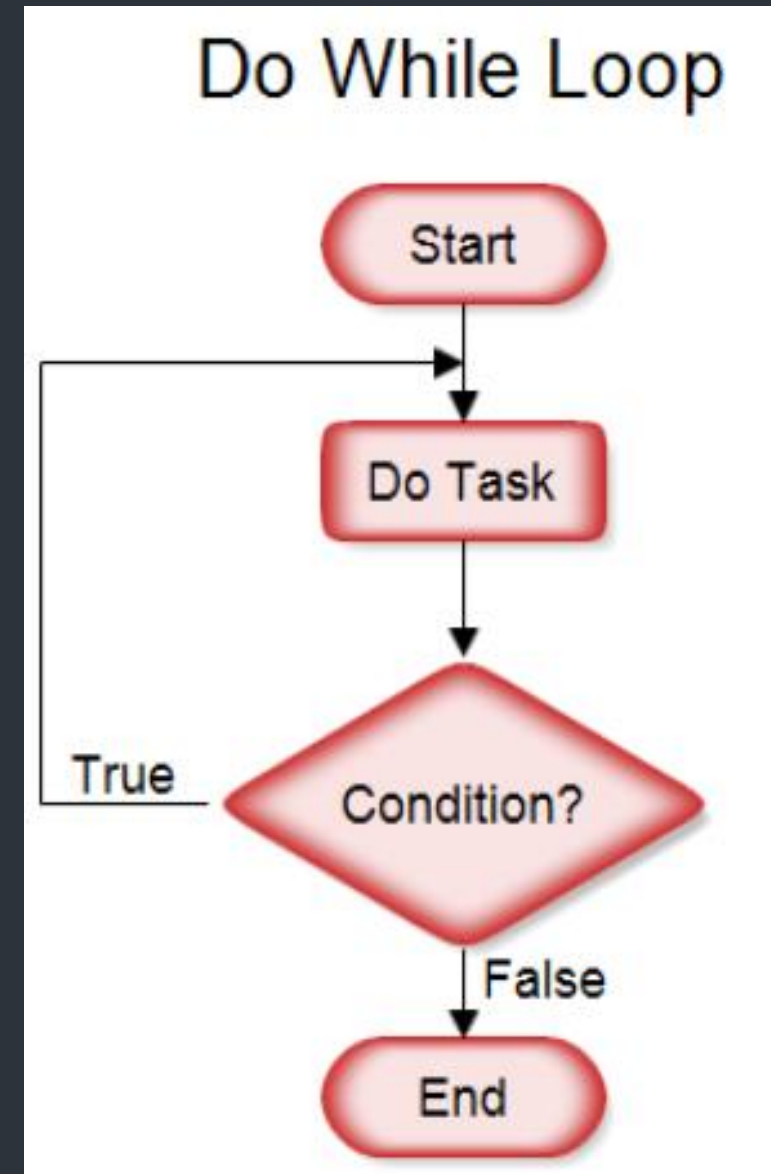
- ▶ Like a while loop, a do-while loop is a loop that repeats while condition is satisfied.
- ▶ Unlike a while loop, a do-while loop tests its condition at the end of the loop.
- ▶ This means that its sequence of activities always runs at least once.

▶ Syntax

Do

Statement

WHILE (condition)



Example: Create the algorithm and the flowchart to Print Hello World 10 times.

➤ Algorithm (using For loop)

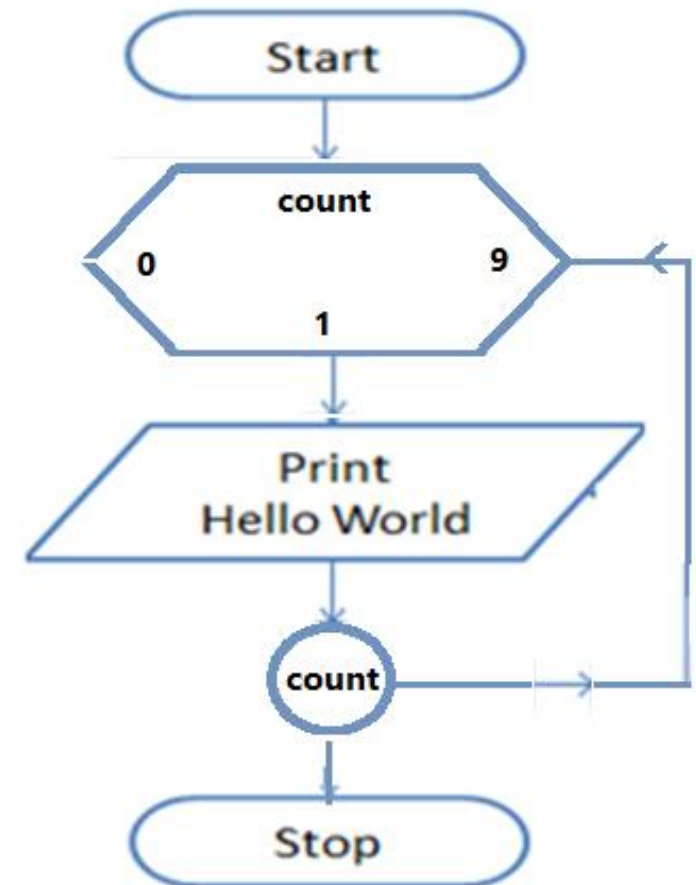
Step 1: Loop: (count = 0 to 9)

Step 2 : Print (Hello World)

Step 3: Loop-end

Step 4: Stop

Flowchart



Example 1: Create the algorithm and the flowchart to Print Hello World 10 times.

► Algorithm (using While loop)

Step 1: Start

Step 2: Initialize count = 0

Step 3: While (count < 10)

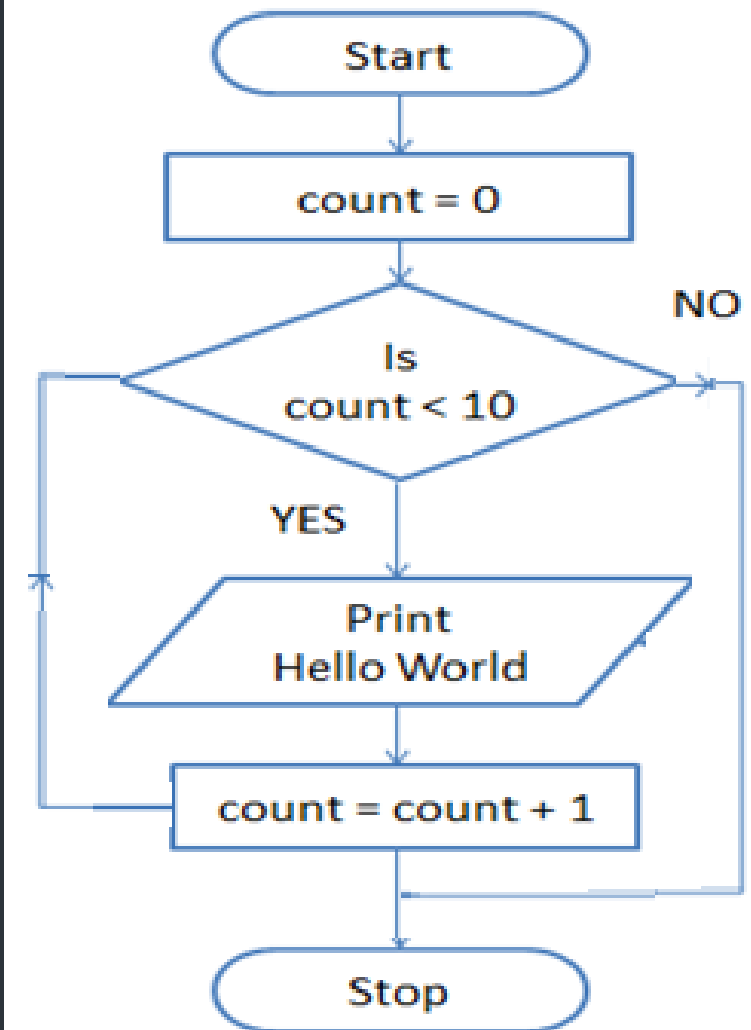
Print (Hello World)

Increment count by 1

While- end

Step 4: stop

Flowchart



Example: Create the algorithm and the flowchart to Print Hello World 10 times.

► Algorithm (using Repeat until)

Step 1: Initialize count = 0

Step 2: **Repeat**

Print (Hello World)

count = count +1

Until (count = 10)

Step 3: stop

► Algorithm (using do- while)

Step 1: Initialize count = 0

Step 2: **do**

Print (Hello World)

count = count +1

While (count < 10)

Step 3: stop

Example 2

أوجد المخطط الانسيابي (Flowchart) بالإضافة إلى كود الشفرة (Pseudo code) لخوارزم يقوم بقراءة درجات طالب في خمس مواد دراسية ثم يقوم بحساب وطباعة مجموع درجات الطالب وكذلك متوسط درجاته

Pseudo code

Variable C, Sum=0, Score: integer

Variable Ave : real

Begin

Loop: (C = 1 to 5)

 Read (Score)

 Sum = Sum + Score

Loop-end: C

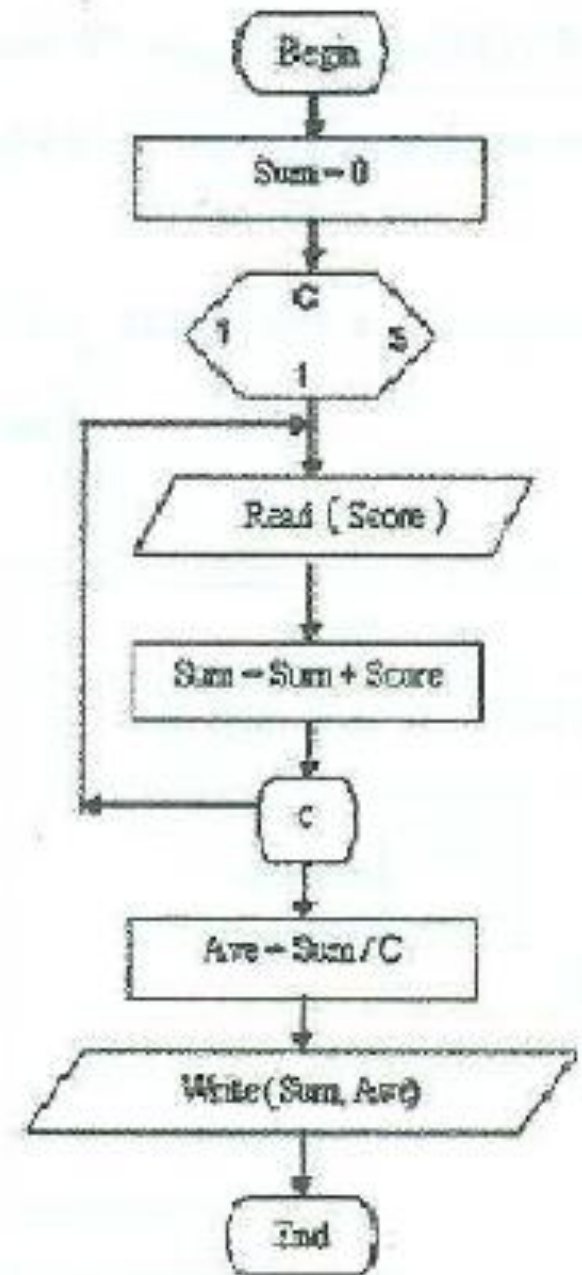
Ave = Sum / C

Write (Sum, Ave)

End

Result of the algorithm

Input : Score	50	70	90	60	80
Output : Sum, Ave	350 70				



Task 1

حساب المعدل التراكمي GPA للطلاب يتم بالتالي:

إذا كان التقدير (grade = A) زاد GPA بمقدار 0.9

إذا كان التقدير (grade = B) زاد GPA بمقدار 0.7

إذا كان التقدير (grade = C) زاد GPA بمقدار 0.5

إذا كان التقدير (grade = F) زاد GPA بمقدار 0.0

أوجد المخطط الانسيابي (flowchart) بالإضافة إلى كود الشفرة (Pseudo code) لخوارزم

يقوم بحساب المعدل التراكمي GPA للطلاب في مادة معينة. إذا تم إدخال تقدير (GPA) خاطئ

فإن الخوارزم ينتهي مع طباعة رسالة "You typed wrong grade". الخوارزم يطبع قيمة

كل من grade and GPA . يمكن اعتبار وجود قيمة أوليه ($GPA = 2.5$)

Task 2

Find Even numbers between 1 to 50.

