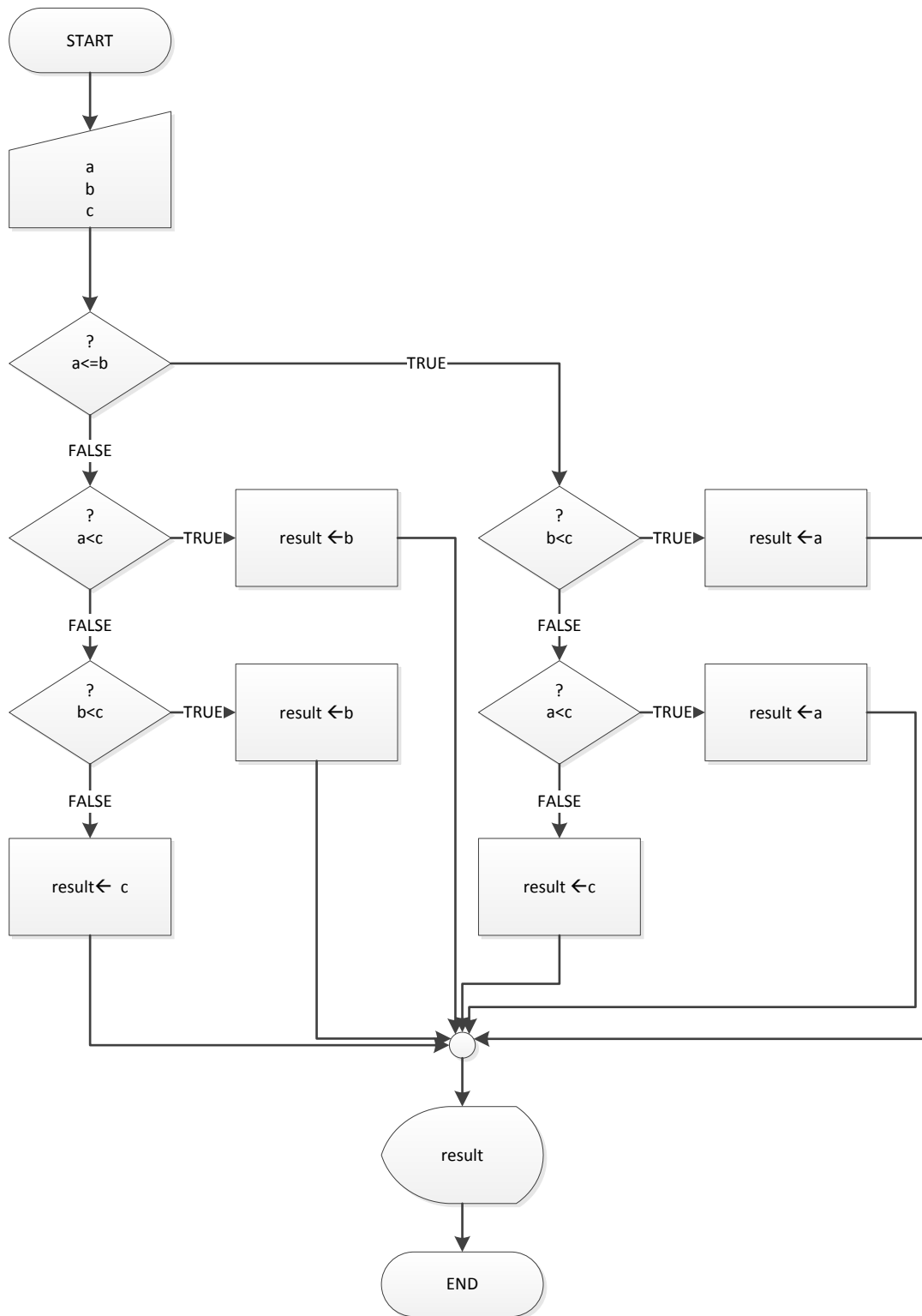Samples

1. **Design a pseudocode that computes x^n. Prompt the user to enter the value of *x* and *n* from keyboard. (25 p)**

    Ex: Sample input for 4 and 3 your design should calculate 4^3 →4*4*4

2. **What is the output of the following flowchart for the values given below: (25 p)**

Flowchart:

START → input (a, b, c) → decision "? a<=b"

- a<=b TRUE → decision "? b<c"
  - b<c TRUE → result ← a
  - b<c FALSE → decision "? a<c"
    - a<c TRUE → result ← a
    - a<c FALSE → result ← c
- a<=b FALSE → decision "? a<c"
  - a<c TRUE → result ← b
  - a<c FALSE → decision "? b<c"
    - b<c TRUE → result ← b
    - b<c FALSE → result ← c

All paths → (merge) → result → END

   a. a=1,b=2,c=3
   b. a=2,b=1,c=3
   c. a=2,b=3,c=1
   d. What does this algorithm do?(Bonus 5p)

3. Consider the pseudocode shown below and then convert the pseudocode to flowchart. (25 p)
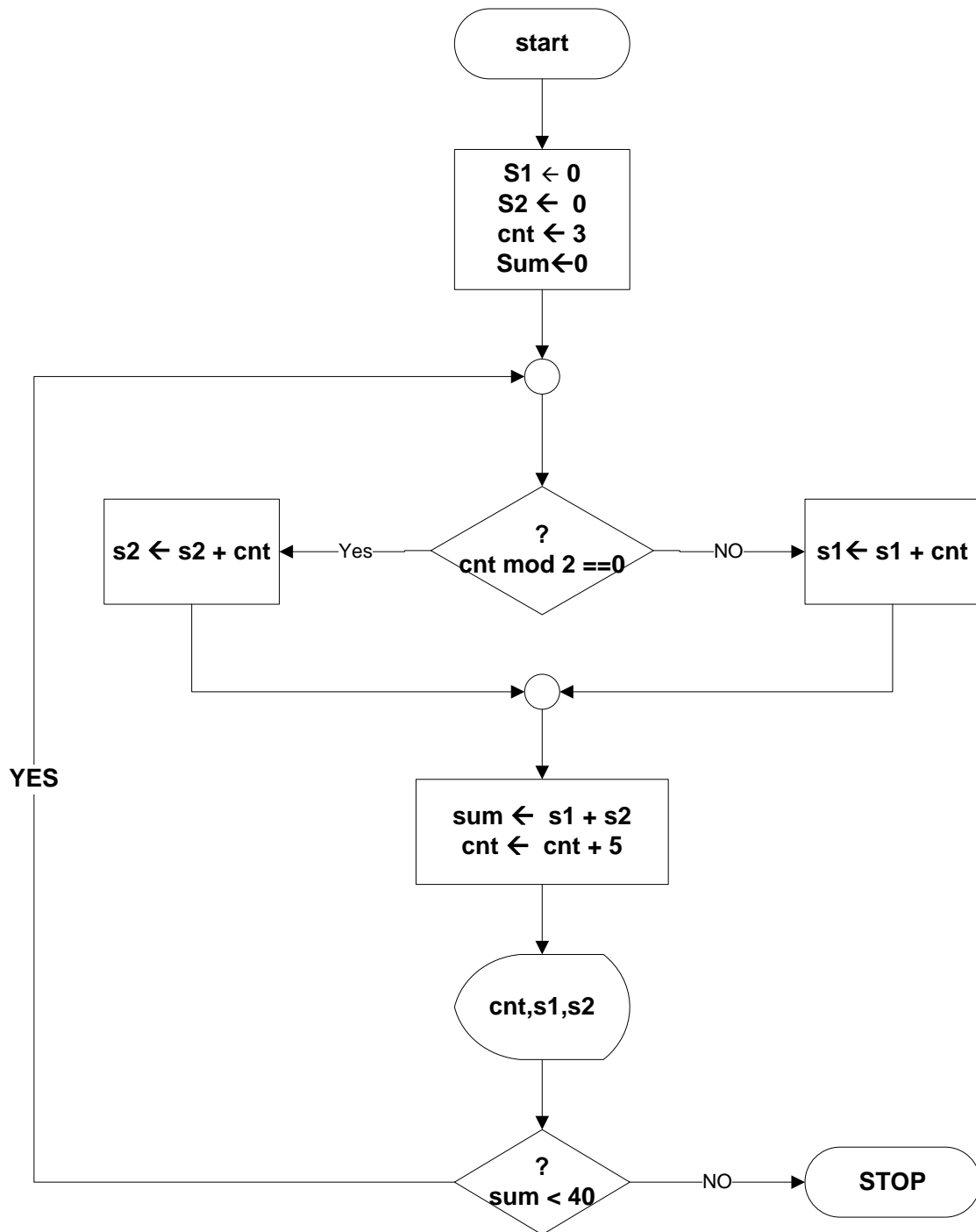
```
DO
DISPLAY "Enter an integer number"
INPUT num
IF  num<100 THEN
    count1=count1+1;
ELSE
    count2=count2+1
ENDIF
WHILE num>0
DISPLAY count1,count2
```

4. Write analgorithm which asks the user to enter their marital status, corresponding to a letter input.

> married = 'm'
>
> single = 's'
>
> divorced = 'd'
>
> widowed = 'w'

When the user enters the letter, their corresponding status should be printed to the screen. If the user enters anything other than *m,s,d,* or *w* the message "Invalid Code" should be printed.

5. Convert the following Flowchart to pseudocode:

```
                              ┌──────────┐
                              │  start   │
                              └────┬─────┘
                                   │
                              ┌────▼─────┐
                              │ S1 ← 0   │
                              │ S2 ← 0   │
                              │ cnt ← 3  │
                              │ Sum←0    │
                              └────┬─────┘
                                   │
                                  ( )◄──────────────────┐
                                   │                    │
                                   ▼                    │
         ┌──────────────┐      ◇───────◇               │
         │ s2 ← s2 + cnt│◄─Yes─│   ?   │─NO─►┌─────────────┐
         └──────┬───────┘      │cnt mod│     │ s1← s1 + cnt│
                │              │2 ==0  │     └──────┬──────┘
                │              ◇───────◇            │
                │                                   │
   YES          └──────────────►( )◄────────────────┘
                                 │
                                 ▼
                         ┌────────────────┐
                         │ sum ← s1 + s2  │
                         │ cnt ← cnt + 5  │
                         └───────┬────────┘
                                 │
                                 ▼
                           ( cnt,s1,s2 )
                                 │
                                 ▼
                             ◇───────◇
                             │   ?   │──NO──►┌────────┐
                             │sum<40 │       │  STOP  │
                             ◇───────◇       └────────┘
```

6. Write an algorithm that inputs a series of 10 numbers, and determines and prints the largest of the numbers.

   Your program should use three variables as follows

   counter: A counter to count to 10 (i.e., to keep track of how many numbers have been input and to determine when all 10 numbers have been processed)

   number: The current number input to the program

   largest: The largest number found so far

7. Generate your solutions using Visio 2000.Draw the Flowchart and Write down the Pseudocode on the same page to calculate and display the following summation value :

Sum = 9 + 11 + 13 + …………+ 57 + 59

8. Using Visio Draw the Flowchart and then write down on the same page the Pseudocode to calculate and display the SUM of the areas of the N circles having the following radii :
5, 10, 15,………………N

Suggested variables : N → will indicate the radius of the last circle in the series

9. Write down the algorithm (Flowchart and Pseudocode) to calculate and display the areas of 10 rectangles. First rectangle will have a length of 2 cm and a width of 1 cm. Each rectangle will have this length increased by 3 and width by 4.

	Length	Width
First rectangle	2	1
Second rectangle	5	5
Third Rectangle	8	9
………..	…..	….

10. Write an algorithm which will get gross_pay as input and then calculate and display the net payment for an employee based on the tax rate given in the table below where;

Net payment= Gross Pay–Tax

Gross Pay	Tax
Less than $2000	0
$2001 to $2500	*Gross pay**0.1
$2501 to $3000	*Gross pay**0.12
$3001 to $3500	*Gross pay**0.14

Above $3500	*Gross pay**0.18

11. **Write a program which will generate the following as output : (use nested loops)**

* * * * *

* * * * *

* * * * *

* * * * *

* * * * *

12. **Write a program which will generate the following as output : (not: use cout and cin statements instead of printf and scanf)**

3 1 1 1 1

2 3 1 1 1

2 2 3 1 1

2 2 2 3 1

2 2 2 2 3

Solution :

```
#include <stdio.h>
main()
{
for (int r=1 ; r<=5 ; r++)
     { printf("\n\n");
          for ( int c=1 ; c<=5 ; c++)
          {  if ( r==c ) printf("3 ");
           else if ( r > c ) printf("2 ");
```

else printf("1 ");

}

}

}

13. **Write down the algorithm (Flowchart and Pseudocode) to calculate the quiz average of N student.**

For solution

Set 0 to sum Input N FOR cnt ←1; cnt<=N;cnt←cnt+1 input quiz sum←sum+quiz ENDFOR Avg←sum / N Display avg	

While Solution

Pseudocode	Flowchart

Set 0 to sum
Input N
Set 1 to cnt
WHILE cnt<=N
     input quiz
     sum←sum+quiz
     cnt←cnt+1
ENDWHILE
avg←sum / N
display avg	

```
        START
          |
       sum<-- 0
          |
          N
          |
       cnt<--1
          |
          O
          |
        ? cnt<=N ----N----> avg<--sum / N
          |Y                      |
        quiz                     avg
          |                       |
   sum<--sum+quiz               STOP
   cnt<--cnt + 1
```
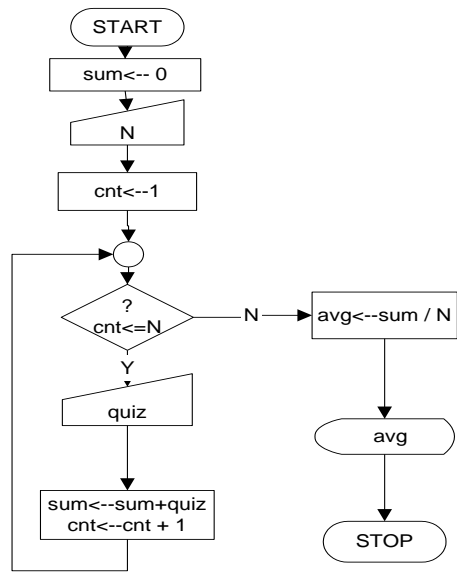
14. Write an algorithm that will produce following multiplication table

a)Use nested while

b)Use nested for

multiplication table

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

a)WHILE Solution
set 1 to r
WHILE r is <=10
   Set 1 to c
    WHILE c is <= 10
    Display r * c
    increase c by 1
   ENDWHILE
Display newline
ENDWHILE
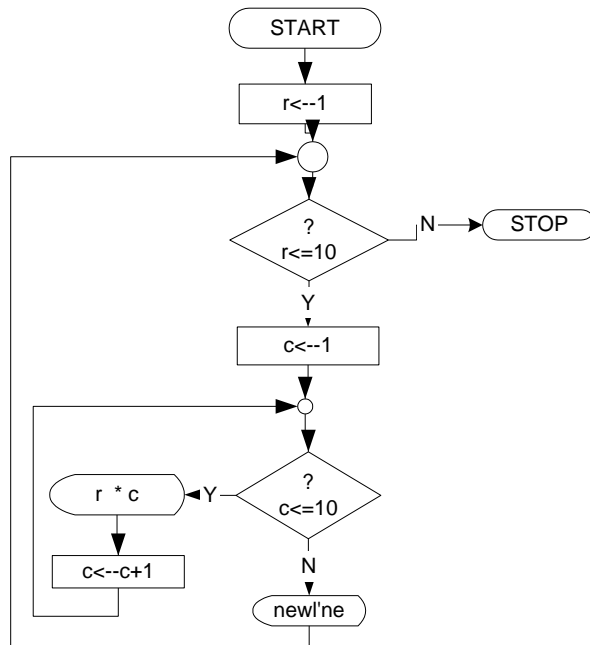
15. Trace the following pseudocodes and then convert them into flowchart:

a)	b)

Set 0 to i
Set 0 to x
  DO
    IF i mod 5 is equal to 0
      x← x + 1
      display x
   ENDIF
  i← i +1
  WHILE i is <= 20
display x
	Set 0 to i
	Set 0 to x
	  FOR i←1; i<10;i←i + 1
	    IF i mod 2 is equal to 1
	      x←x+1
	    ELSE
	      x←x - 1
	    ENDIF
	  Display x
	ENDFOR

16.

a) Trace the flowchart given below for num=3.

      b) Show the *final* Output.

```
                    ┌──────────────┐
                   (    Start       )
                    └──────────────┘
                          │
                          ▼
                    ┌──────────────┐
                    │  row ← 1     │
                    │  col ←  1    │
                    │  num ← 0     │
                    └──────────────┘
                          │
                          ▼
                    ┌──────────────┐
                    │     num      │
                    └──────────────┘
                          │
                          ▼
                         (○)◄─────────────────────────────┐
                          │                                │
                          ▼                                │
   No               ◇  ?                                   │
◄─────────────    row ≤ num   ───Yes──┐                    │
                         ◇            │                    │
                                      ▼                    │
                                     (○)◄──────────┐       │
                                      │            │       │
                                      ▼            │       │
                              ◇   ?                │       │
                            col ≤ num ───Yes──┐    │       │
                                 ◇            │    │       │
                                 │            ▼    │       │
                                No      ◇   ?      │       │
                                 │   No row≤ col Yes       │
                                 │  ┌──◇      ◇──┐  │       │
                                 │  ▼            ▼  │       │
                                 │ ('X')        ('+')│       │
                                 │  │            │  │       │
                                 │  └────►(○)◄───┘  │       │
                                 │         │        │       │
                                 ▼         ▼        │       │
                        ┌──────────┐ ┌──────────┐   │       │
                        │ col← 1   │ │col ← col+1│   │       │
                        │ row←row+1│ └──────────┘   │       │
                        └──────────┘      │         │       │
                             │            └─────────┘       │
                             ▼                              │
                        ( Begin a )                         │
                        ( New Line )                        │
                             │                              │
   ┌─────────┐               └──────────────────────────────┘
  ( Stop     )
   └─────────┘
```
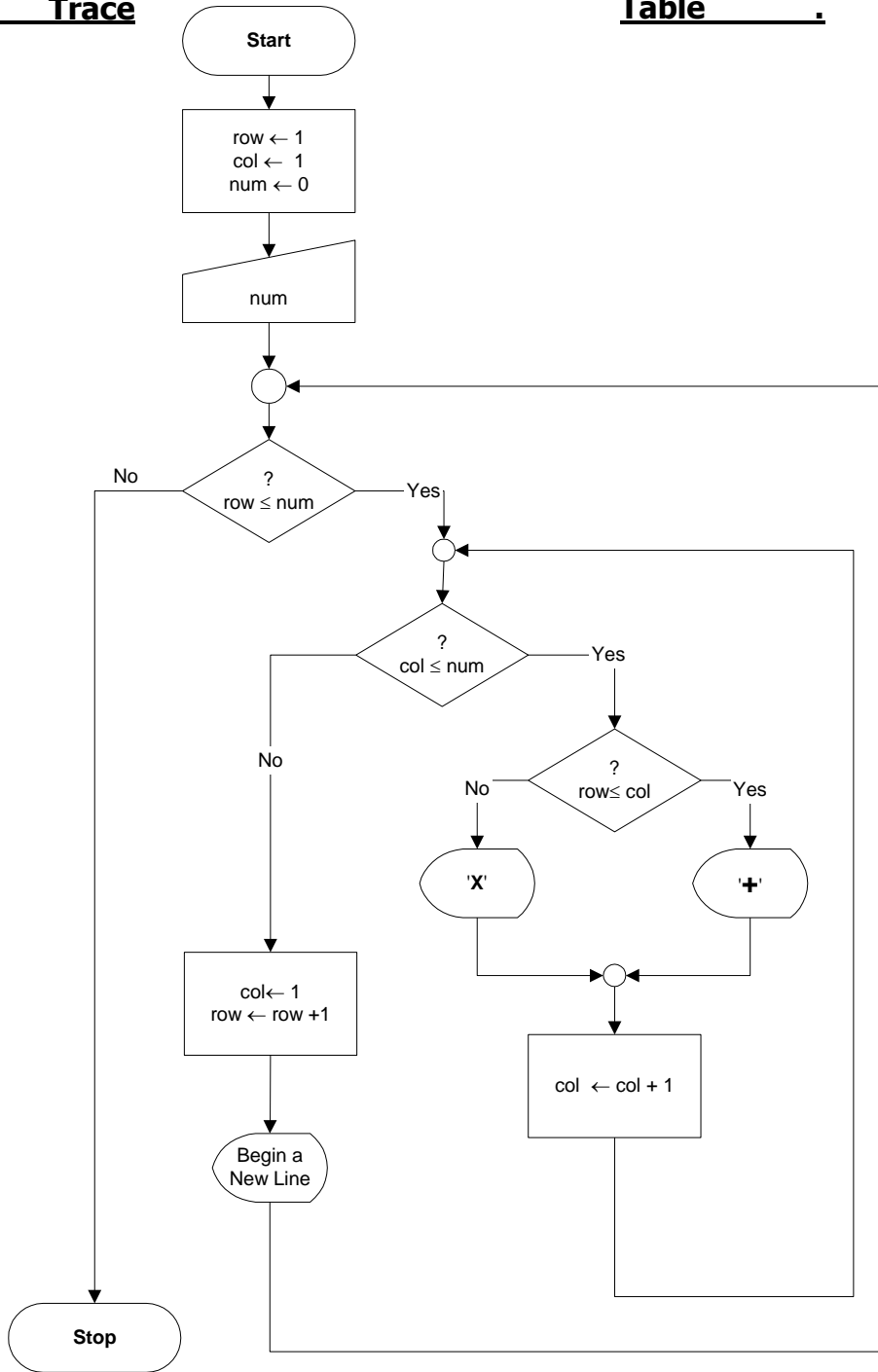
17.

a)Assuming N is 60 write down the "Trace Table" and the "Output" for the following Flowchart:(20 points)

b)Write down Pseudocode for following flowchart.
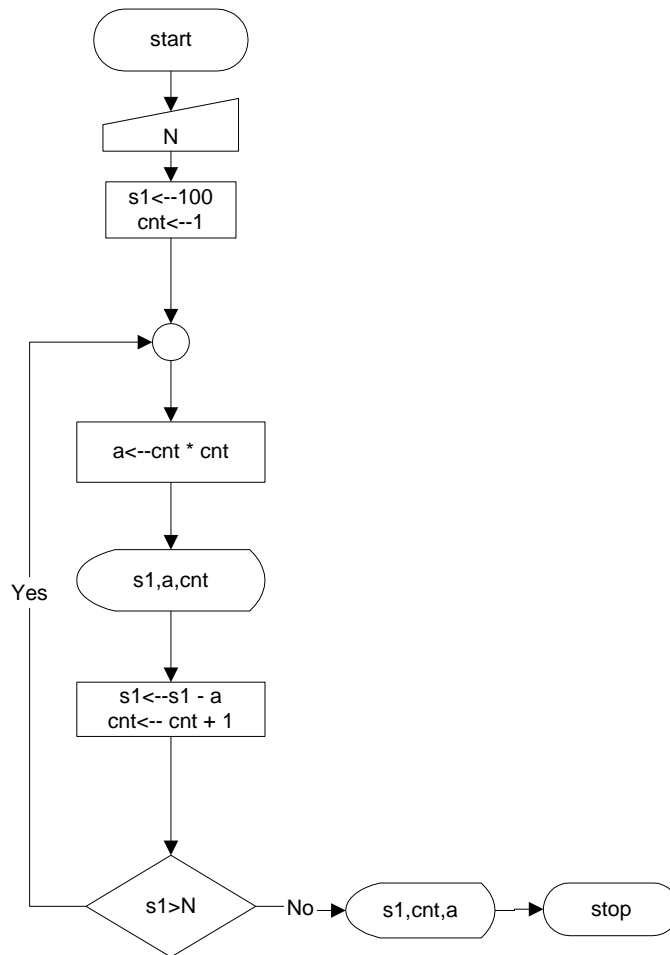
Solution:

N	S1	cnt	a	display
60	100	1	1	100 1 1
	99	2	4	99 4 2
	95	3	9	95 9 3
	86	4	16	86 16 4

18. **Write down the equivalent pseudocode for the following flowchart:** (20 points)

```
                    ┌─────────────┐
                    │    START    │
                    └─────────────┘
                           │
                           ▼
                  ┌──────────────────┐
                  │    eno<--1       │
                  │    tax<--0       │
                  │   salary<--0     │
                  └──────────────────┘
                           │
                           ▼
                          ( )◄────────────────────────────────┐
                           │                                   │
                           ▼                                   │
                    ◇ ecode='A' ◇                              │
         F ─────────┘           └───────── T                   │
         │                                 │                   │
         ▼                                 ▼                   │
    ◇    ?    ◇                       ◇    ?    ◇              │
  F─┘ hcode='U' └─T               F─┘ scode='Y' └─T           │
  │              │                │               │           │
  ▼              ▼                ▼               ▼           │
┌──────────┐ ┌──────────┐   ┌──────────┐  ┌──────────┐       │
│bonus<--200│ │bonus<--100│   │bonus<--10 │  │bonus<--0  │     T
└──────────┘ └──────────┘   └──────────┘  └──────────┘       │
     │            │              │              │             │
     └─────►( )◄──┘              └─────►( )◄────┘             │
            │                           │                     │
            ▼                           │                     │
       ◇    ?    ◇                      │                     │
     F─┘ pcode='Z' └─T                  │                     │
     │            │                     │                     │
     ▼            ▼                      │                     │
┌──────────┐ ┌──────────┐               │                     │
│salary<--500│ │salary<--600│            │                     │
└──────────┘ └──────────┘               │                     │
     │            │                      │                     │
     └─────►( )◄──┘                      │                     │
            │                            │                     │
            └──────────►( )◄─────────────┘                     │
                         │                                     │
                         ▼                                     │
                 ┌──────────────┐                              │
                 │  eno<--eno+1 │                              │
                 └──────────────┘                              │
                         │                                     │
                         ▼                                     │
                   ◇    ?    ◇─────────────────────────────────┘
                   │ eno<=10  │
                   └──────────┘
                         │ F
                         ▼
                  ┌─────────────┐
                  │    STOP     │
                  └─────────────┘
```
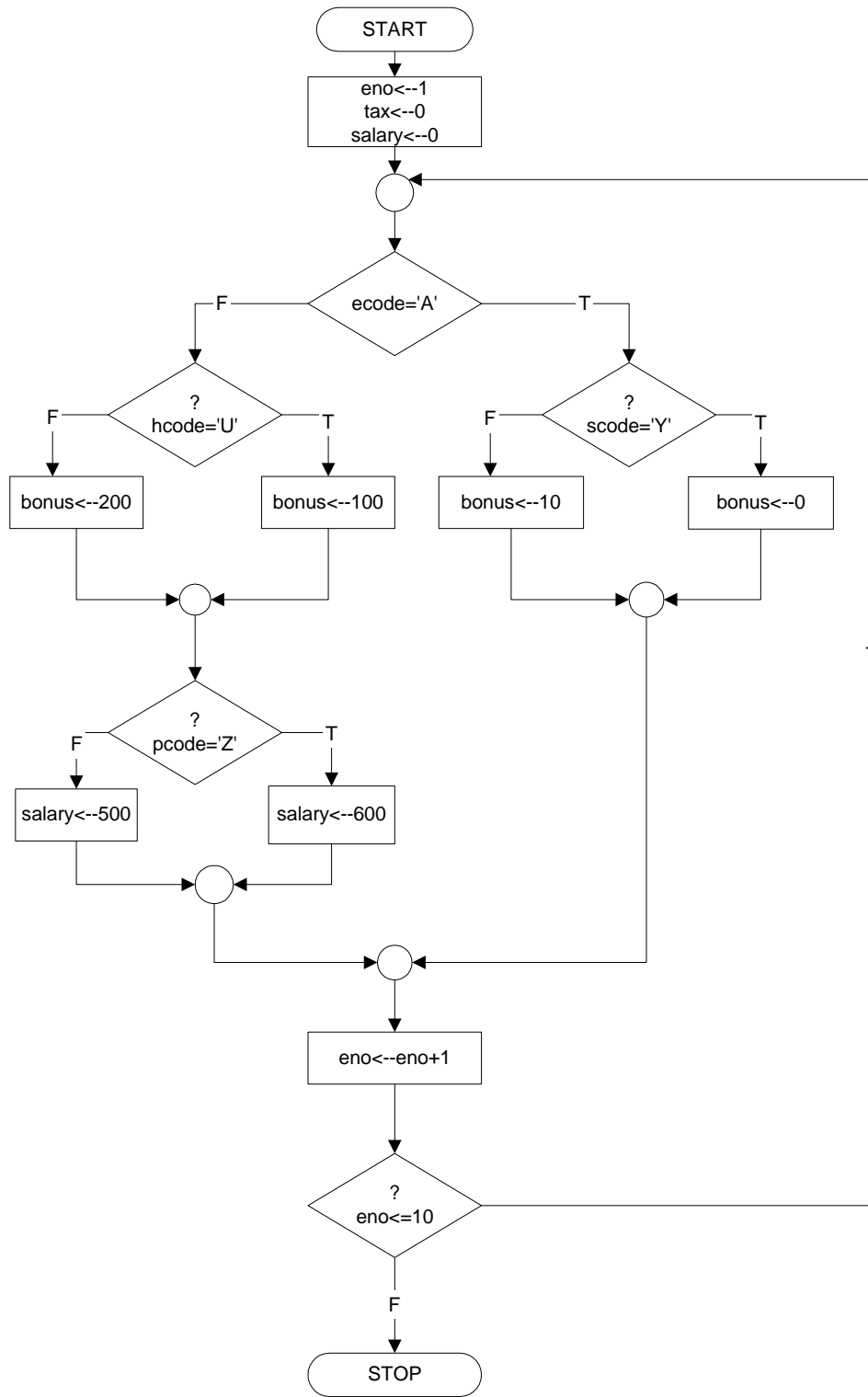
19. Write an **algorithm** and draw a **flowchart** that will calculate the bookstore weekly payroll. The inputs to the **algorithm** are the <u>number of hours</u> the employee works in one week and the employee's <u>hourly pay rate</u>. The **algorithm** is to calculate the employee's gross pay including the possibility of overtime pay. Regular pay is the <u>number of hours</u> the employee worked (up to 40 hours) times the <u>hourly pay rate</u>. The program will allocate overtime payment if the employee has worked more than 40 hours. The overtime pay equals the <u>number of hours</u> the employee worked over 40 hours, multiplied by 1.5 (the overtime pay factor) ,multiplied by the <u>hourly pay rate</u>. The program should output the worker's regular pay, over-time-pay, gross pay (The sum of the regular pay and overtime pay) AND TOTAL PAYROLL(totpay←totpay+grosspay). The algor'thm should prompt to user to enter the data for the next employee .When the user response that there are no more employees to process the design should display no of employees and total payroll for bookstore.

Regular pay → is the payment for 40 hours or less work

Over-time-pay → is the payment for the employee work above 40 hours

Gross pay = regular pay + over-time-pay

Answer

Read(Input) nofhours and pay_rate

If nofhours is less than or equal to 40 then

    Calculate regular_pay by multiplying nofhours with pay_rate set it to the regular_pay.

    Set 0 to overtime_pay
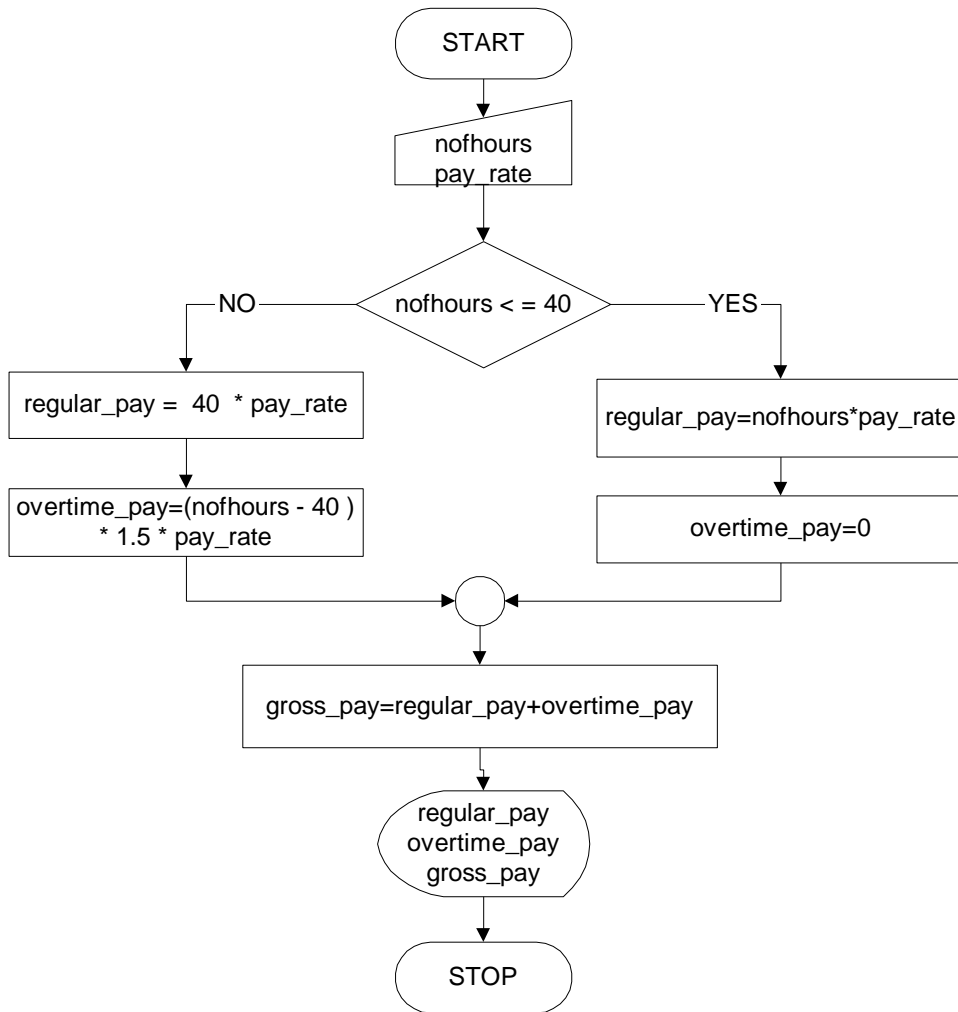
Else

    Calculate regular_pay by multiplying 40 with pay_rate set it to the regular_pay.

    Calculate overtime_pay by nofhours minus 40 multiply with 1.5 and pay_rate,
   set it to the overtime_pay.

Endif

Calculate gross_pay by adding regular_pay to overtime_pay

Display regular_pay, overtime_pay, gross_pay.

```
                        ┌─────────────┐
                        │    START    │
                        └──────┬──────┘
                               │
                        ┌──────┴──────┐
                        │  nofhours   │
                        │  pay_rate   │
                        └──────┬──────┘
                               │
                        ◇─────────────◇
              NO ───────  nofhours < = 40  ─────── YES
              │         ◇─────────────◇              │
              │                                        │
    ┌─────────┴──────────┐              ┌──────────────┴──────────────┐
    │ regular_pay = 40   │              │ regular_pay=nofhours*pay_rate│
    │       * pay_rate   │              └──────────────┬──────────────┘
    └─────────┬──────────┘                             │
    ┌─────────┴──────────┐              ┌──────────────┴──────────────┐
    │overtime_pay=(nofhours - 40)│      │       overtime_pay=0         │
    │    * 1.5 * pay_rate │              └──────────────┬──────────────┘
    └─────────┬──────────┘                             │
              │              ◯                          │
              └──────────────┤────────────────────────┘
                             │
              ┌──────────────┴──────────────┐
              │ gross_pay=regular_pay+overtime_pay │
              └──────────────┬──────────────┘
                             │
                     ┌───────┴────────┐
                     │  regular_pay   │
                     │  overtime_pay  │
                     │   gross_pay    │
                     └───────┬────────┘
                             │
                      ┌──────┴──────┐
                      │    STOP     │
                      └─────────────┘
```

20. **Write down the "Trace Table" and the "Output" and then the "Pseudocode" for the following Flowchart :**
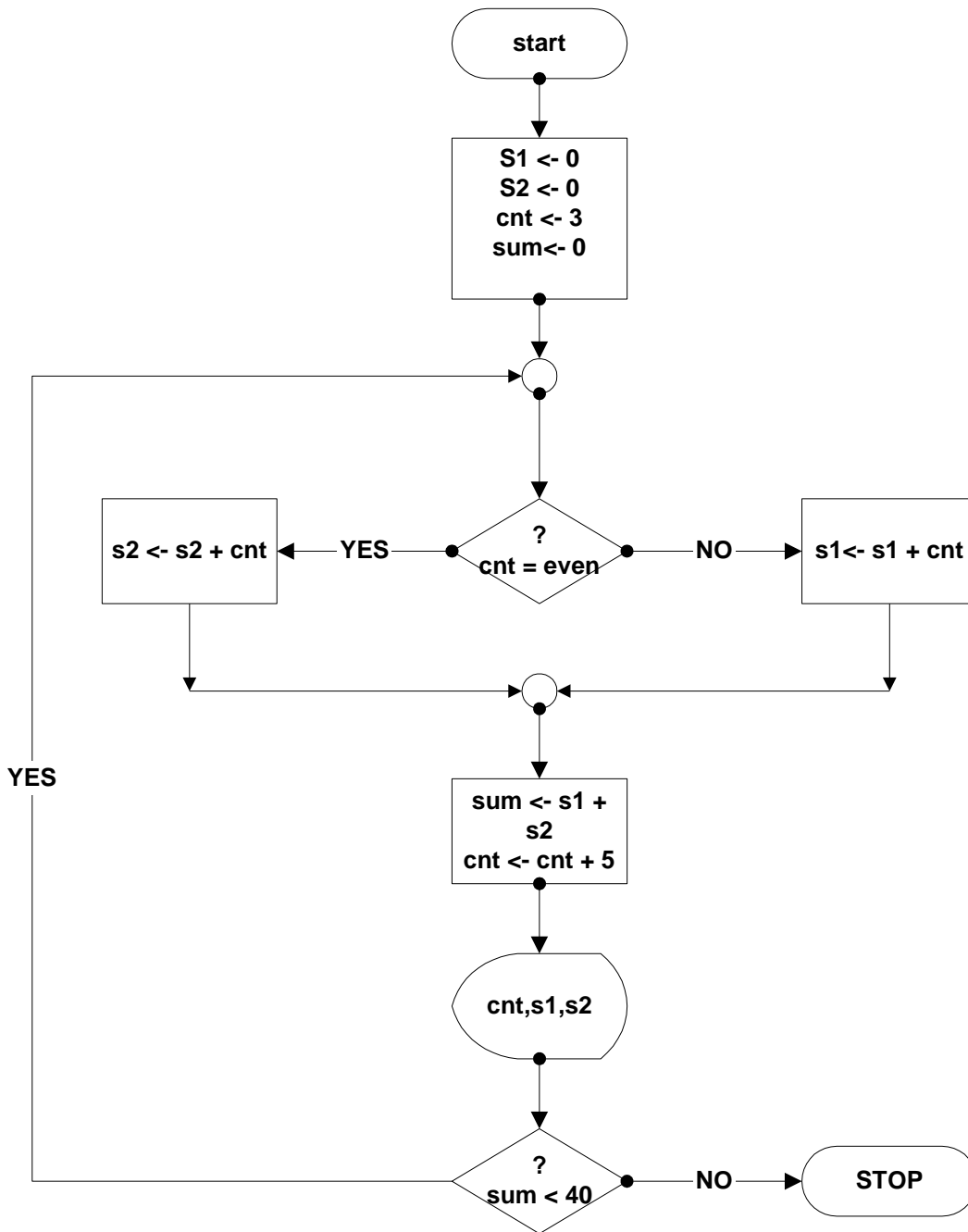
Tracetable :   **Output :**

Set s1 to 0
Set s2 to 0
Set cnt to 3
Set sum to 0
DO
      IF cnt is an even number
            Add cnt to s2
      ELSE
            Add cnt to s1
      ENDIF

Add s1+s2 and set it to sum
Increase cnt by 5
Display cnt, s1, s2
WHILE sum is less than 40

21. Write down the "Trace Table" and the "Output" for the following "Pseudocode" :

Set s1 to 10

Set s2 to 100

Set cnt to 2

WHILE cnt is less than 5

       Calculate cnt*cnt*cnt and set it to a

       Add a to s1

       Calculate cnt*cnt and subtract from s2

       Calculate s1+s2 and set it to sum

       Display sum

       Increase cnt by 1

ENDWHILE

Tracetable :

s1	s2	cnt	a	sum
~~10~~	~~100~~	~~2~~	~~?~~	~~?~~
~~18~~	~~96~~	~~3~~	~~8~~	~~114~~
~~45~~	~~87~~	~~4~~	~~27~~	~~132~~
109	71	5	64	180

Output :

114

132

180