ASSIGNMENT 1

TOPIC 1-3

GROUP MEMBER'S (MATRIC NO):

SECTION:

QUESTION 1:

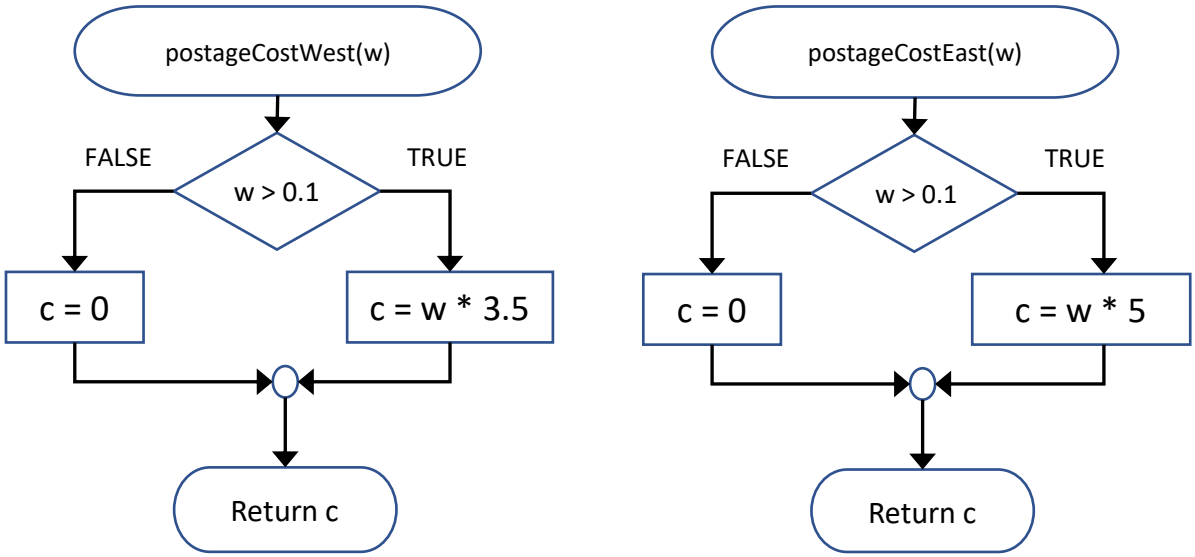
You are requested by the client to write a program to sum the total postage cost of items based on the location. Below is the list of requirements given by the client.

- The user can enter the number of item to be processed by the program (not more than 10 items)
- The user must choose the location to send; West Malaysia or East Malaysia.
- The program should able to detect if item's weight entered is a valid one. Item should not weight more than 5 Kg. A zero (0) or less value entry is also considered as an invalid input of item weight.
- Two Functions, postageCostWest(weight) and postageCostEast(weight) has been prepared to calculate the postage cost to be charged to the item. Based on its weight, item may not be charged thus this function may return a zero (0) value of postage cost.
- The final outputs of the program are the total number of item being charged with postage cost and the sum of the overall postage cost.

Draw a flow chart to solve the above problem. Below is the list of identifiers you have to use in your flow chart.

Identifiers	Descriptions
item_no	Number of item to be processed
count	Count the number of items being processed.
Weight	Weight of item entered by the user.
Location	Location to send; West Malaysia or East Malaysia.
cost	Postage cost of item calculated and returned by postageCost(weight) function.
item_charged	Total number of items being charged with postage cost.
sum_cost	Sum of overall cost for items being charged with postage cost.

You need to include a call to functions in your flow chart. Flow chart for postageCostWest(weight) function and postageCostEast(w) function has been prepared as follows:



*Use on-page connector in your flowchart design
*Design your flowchart in MSWord or other tools, convert to PDF before submission.

QUESTION 2:

1. Read thoroughly the question below. Produce a pseudo code and flow chart for solving the problem.
2. Write a C ++ program to calculate the net salary for an employee of a company that manufactures children's toys.

The company has three types of employees:
G - fixed paid employees; K- Contract workers; and S - subcontract workers.
The input to this problem consists of the employee name, employee number, and employee code: G, K or S. Follow -up input depends on the value of the employee code and the associated category.
Fixed paid employees will be paid a fixed amount of payment at the end of each working month. Fixed -paid employees can be categorized under P: manager or B: non -manager.
Only non -managerial category employees are allowed to claim overtime pay. For the first 10 hours of overtime, employees will be paid at RM15 per hour. For each subsequent hour, employees will be paid at RM12 per hour. However, in a month an employee can claim payment of up to 20 hours of overtime. Claims for payment for excessive hours of time will be rejected. For this category net salary is calculated as: fixed salary + overtime salary.
Contract workers will be paid according to the number of hours worked and based on category: B- Recovery; S- Maintenance
Recovery work will be paid an average of RM20 per hour with claims of up to 100 hours.
Maintenance work will be paid at RM10 per hour for the first 50 hours and RM5 per hour for the next hour. The maximum amount that can be claimed is also limited to 100 hours.
Subcontract workers will be paid according to the number of toys assembled. Subcontract employees can only assemble toys from one of the categories:
B- Large size toy; S- Medium size toy; K- Small size toy
Each large size toy successfully assembled will be paid RM8 each; medium size toy will be paid RM5 each, and small size toys will be paid RM2 each.
At the end of the process display the employee's name, employee number and income for the month. Use an appropriate selection control structure to each of the previously described cases. Test your program to make sure every condition is correctly executed.

LAB ASSIGNMENT 1

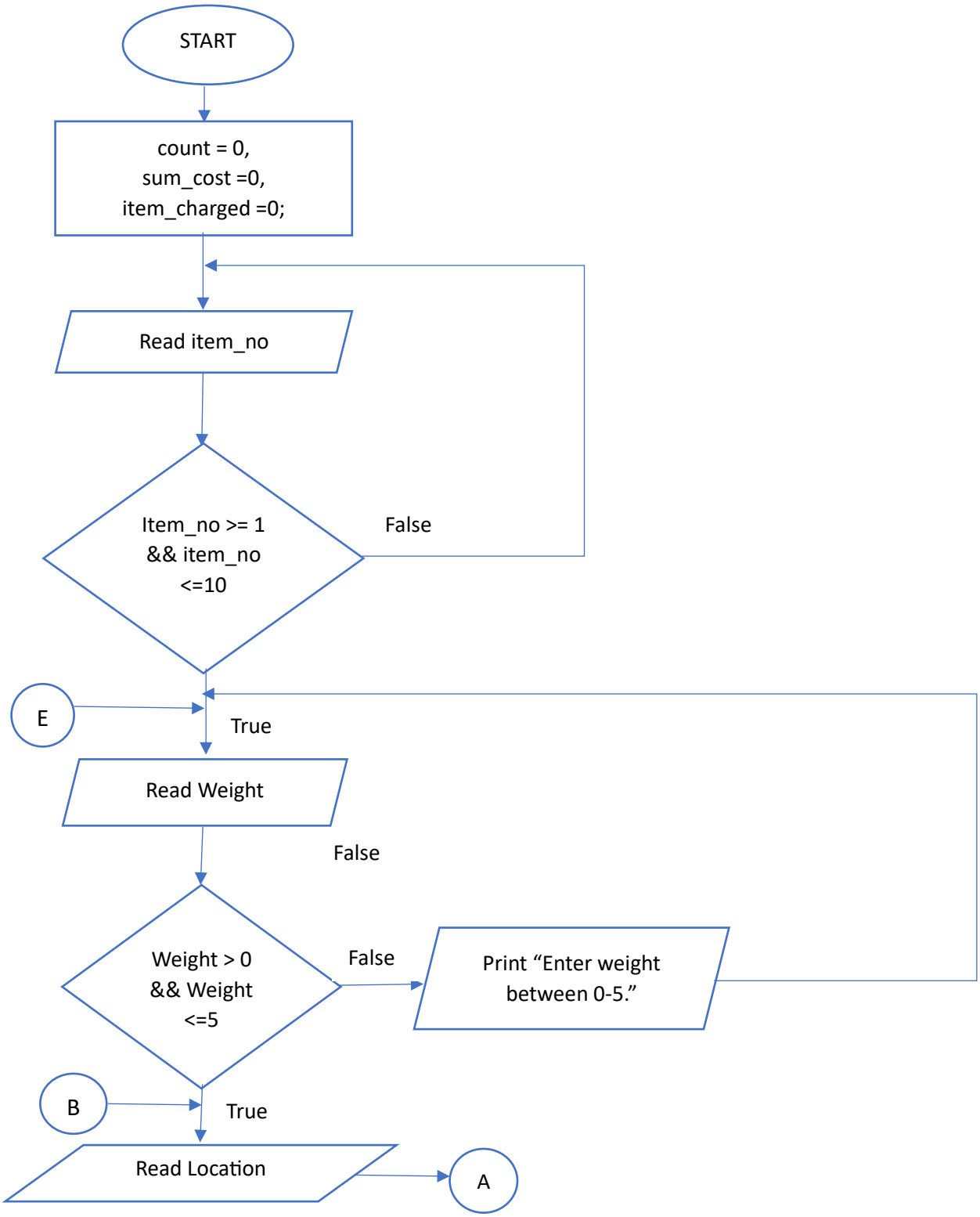
NAME: AMAN SUFIAN SHAH BIN SHAMSUDDIN

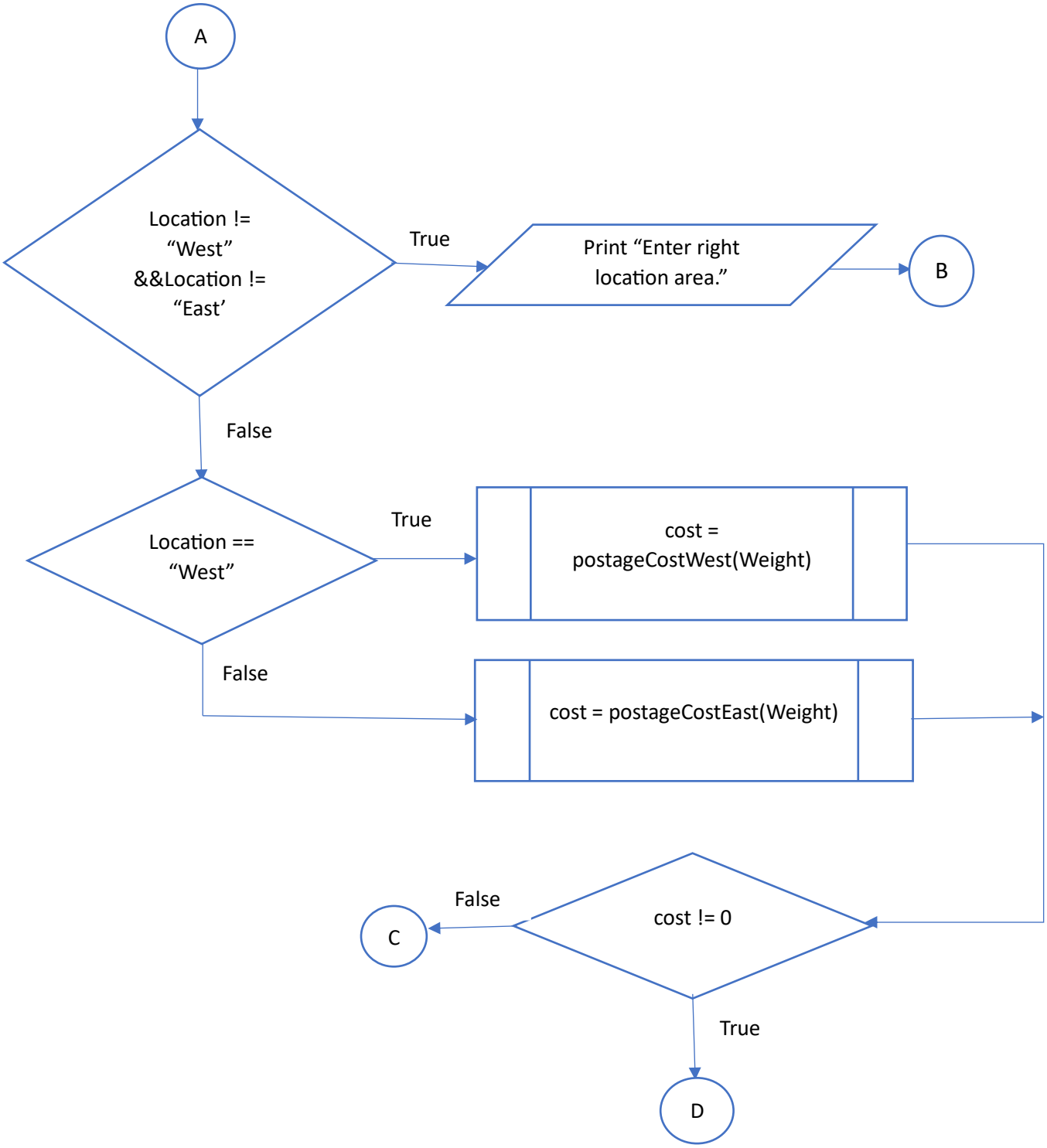
MATRIC NO. : A24CS0046

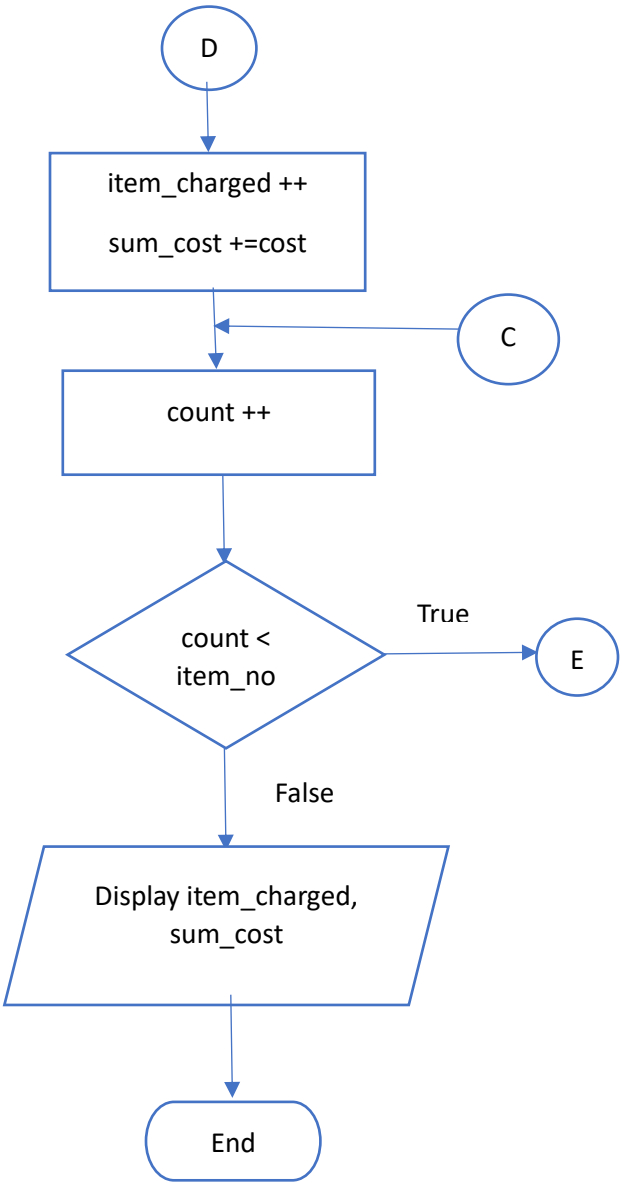
NAME: IDA YATULLAILIYEH BINTI AMRUN

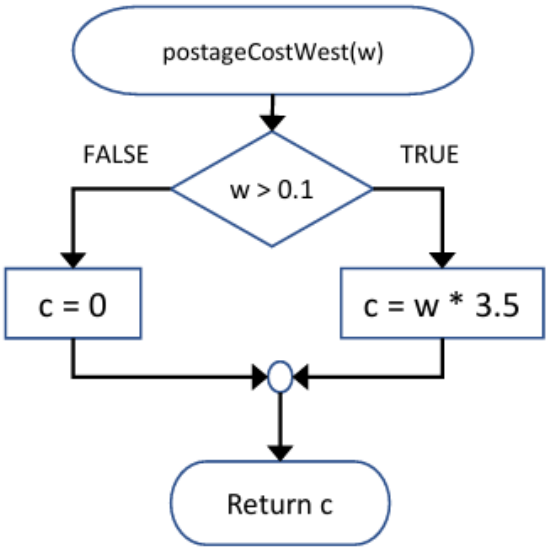
MATRIC NO. A24CS0084

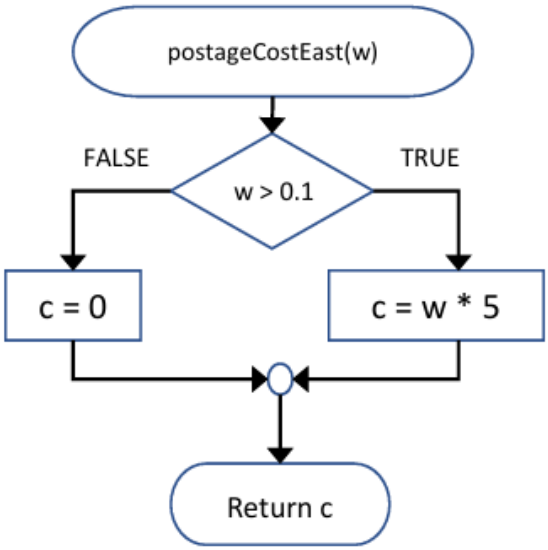
QUESTION 1




A
Location != "West" &&Location != "East'
True
Print "Enter right location area."
B
False
Location == "West"
True
cost = postageCostWest(Weight)
False
cost = postageCostEast(Weight)
cost != 0
False
C
True
D


D
item_charged ++
sum_cost +=cost
C
count ++
count <
item_no
True
E
False
Display item_charged,
sum_cost
End


postageCostWest(w)
FALSE
w > 0.1
TRUE
c = 0
c = w * 3.5
Return c


postageCostEast(w)
FALSE
w > 0.1
TRUE
c = 0
c = w * 5
Return c

QUESTION 2

PSEUDOCODE:

1. Start
2. Set salary_nonmanager = 1500, salary_manager = 2500
3. Read employee_name, employee_number
4. Calling employee_code = getCode()
5. if (employee_code == 'G')
    - 5.1 Calling Status = getStatus()
        - 5.1.1 If (Status =='P')
            - 5.1.1.1 net_salary= salary_manager
        - 5.1.2 else
            - 5.1.2.1 Calling overtime = getOver()
            - 5.1.2.2 net_salary = salary_nonmanager + overtime
6. else if(employee_code == 'K')
    - 6.1 Calling category =getCategory()
        - 6.1.1 if (category == 'B')
            - 6.1.1.1 Calling net_salary=getSalaryB()
        - 6.1.2 else
        - 6.1.2.1 Calling net_salary = getSalaryS()
7. else
    - 7.1 Read sizetoys
    - 7.2 while (sizetoys == 'B' && sizetoys == 'S' && sizetoys == 'K')
        - 7.2.1 Read amount
        - 7.2.2 while (amount > 0)
            - 7.2.2.1 if (sizetoys =='B')
                - 7.2.2.1.1 net_salary = 8* amount
            - 7.2.2.2 else if(sizetoys=='S')
                - 7.2.2.2.1 net_salary =5*amount
            - 7.2.2.3 else
                - 7.2.2.3.1 net_salary = 2*amount
            - 7.2.2.4 end if
        - 7.2.3 end while
    - 7.3 end while
8. end if
9. Print employee_name, employee_number, net_salary
10. End

getCode()

- Start
- Read c
- while(c !='G'&& c!= 'K' && c!='S')
  -Read c
- End while
- Return c
- End

getCategory()

- Start
- Read k
- while(k !='B'&& k!= 'S')
  -Read k
- End while
- Return k
- End

getStatus()

- Start
- Read s
- while(s !='P'&& s!= 'B')
  - Read s
- End while
- Return s
- End

getOver()

- Start
- Read h
- while (h<0)
    - Read h
- End while
- if(h>=0 && h<=10)
    - timepay = h*15
- else if (h>10 && h<=20)
    - timepay = 150 + (h-20)*12
- else
    - timepay =270
- End If
- Return timepay
- End

getSalaryB()

- Start
- Read masa
- while(masa <0)

-Read masa

- End while
- If (masa <= 100)

-Salary = masa *20

- Else

-Salary = 100*20

- End If
- Return Salary
- End

getSalaryS()

- Start
- Read masa
- While (masa <0)

-Read masa

- If (masa >= 0 && masa <= 50)

-Salary = masa *10

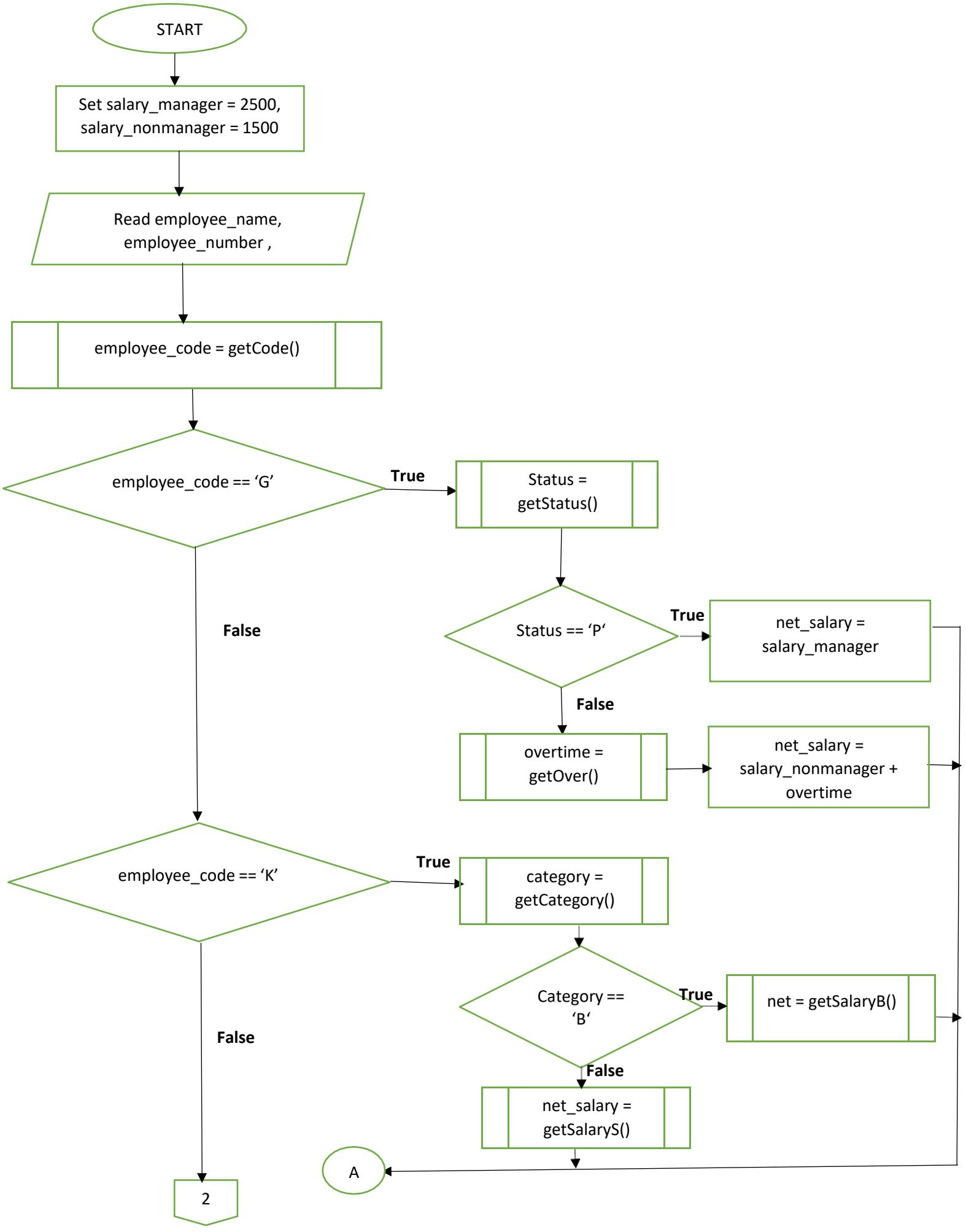
- else if(masa<=100)

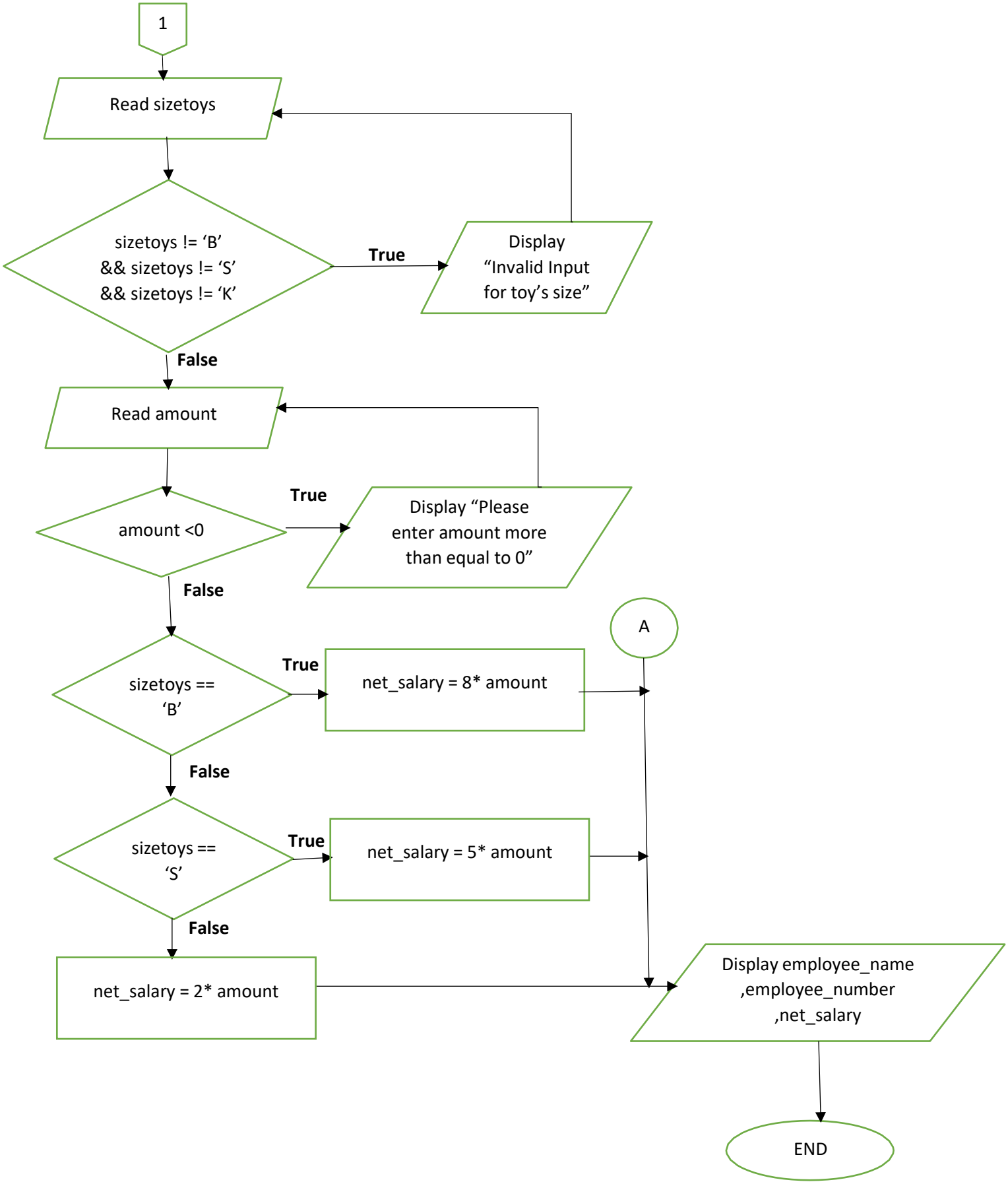
-Salary = 500 + (masa -50)*5

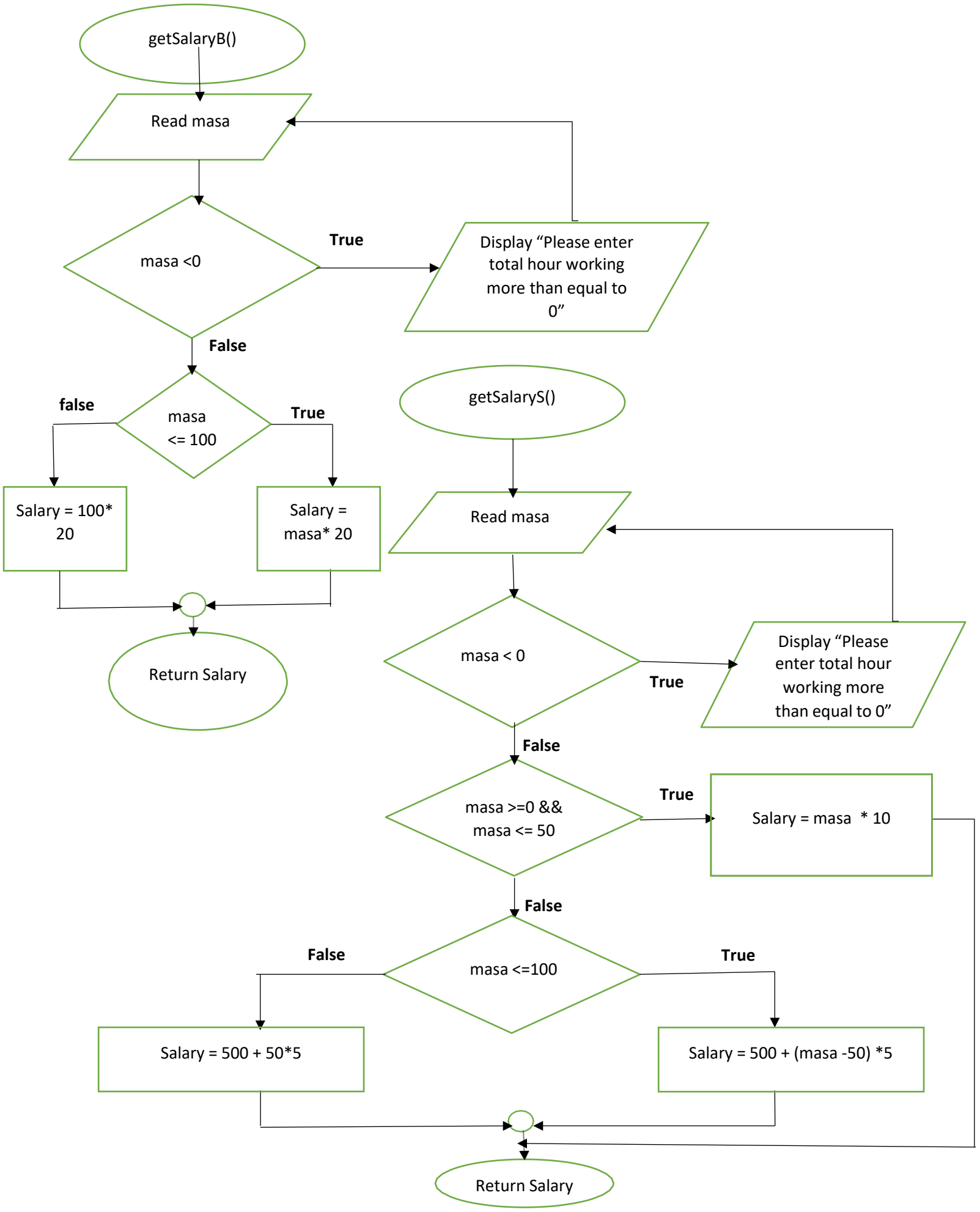
- else

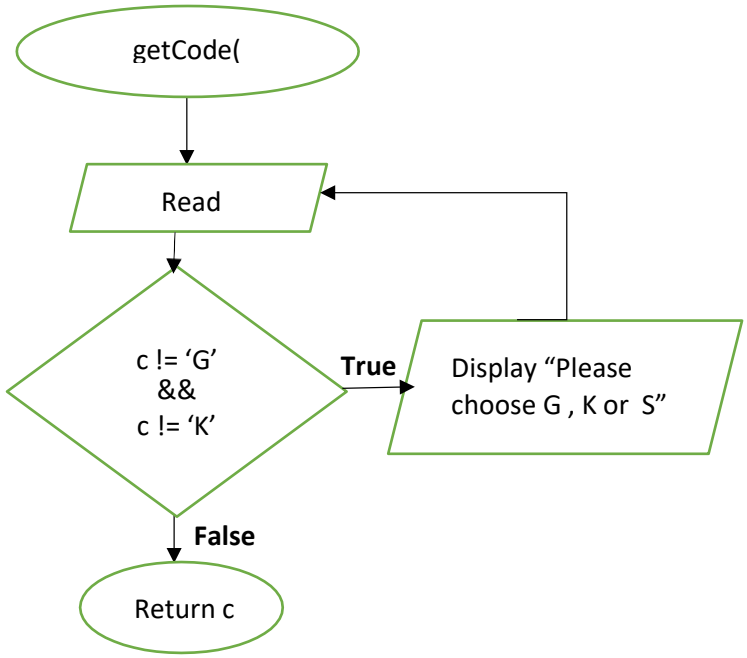
-Salary =750

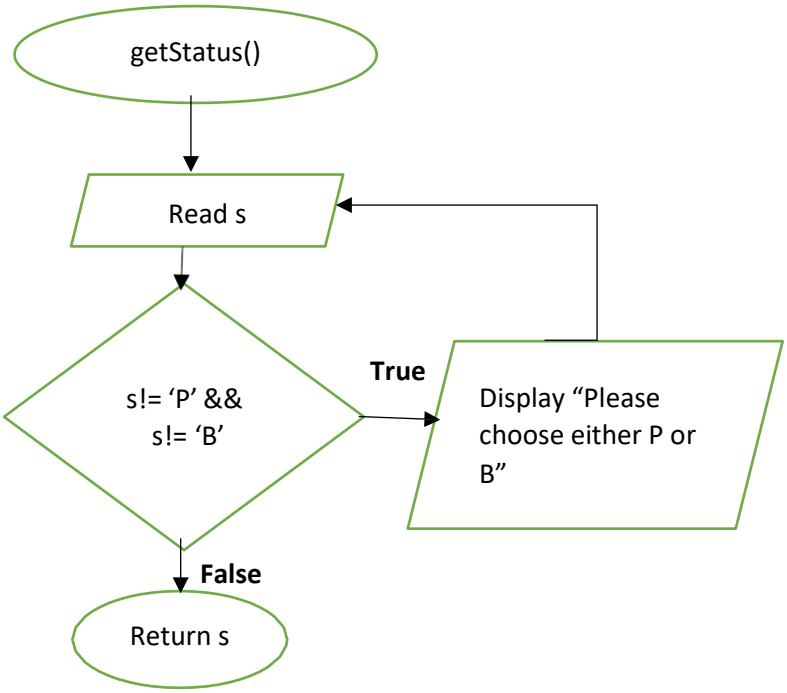
- End If
- Return Salary
- End

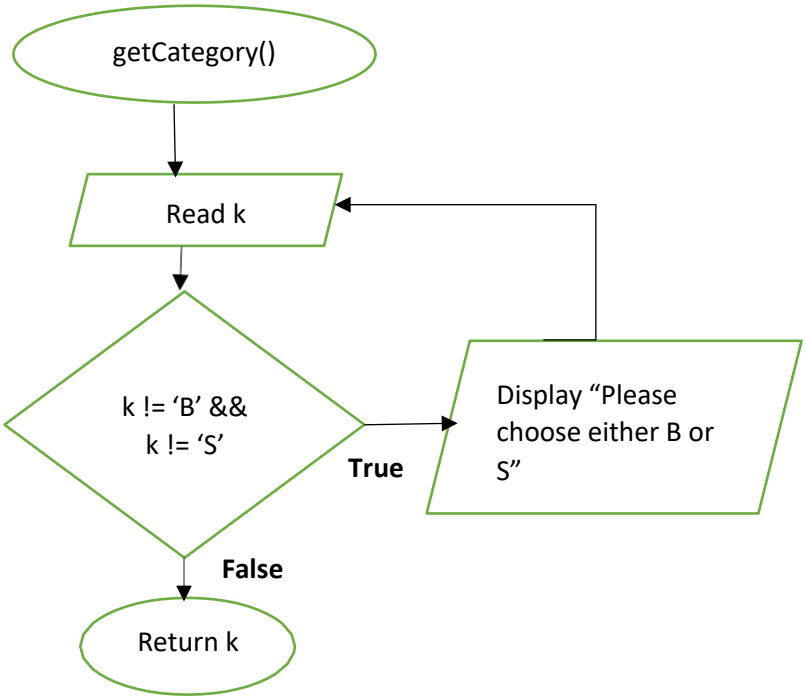

START
Set salary_manager = 2500, salary_nonmanager = 1500
Read employee_name, employee_number ,
employee_code = getCode()
employee_code == 'G'
True
Status = getStatus()
Status == 'P'
True
net_salary = salary_manager
False
overtime = getOver()
net_salary = salary_nonmanager + overtime
False
employee_code == 'K'
True
category = getCategory()
Category == 'B'
True
net = getSalaryB()
False
net_salary = getSalaryS()
False
A
2

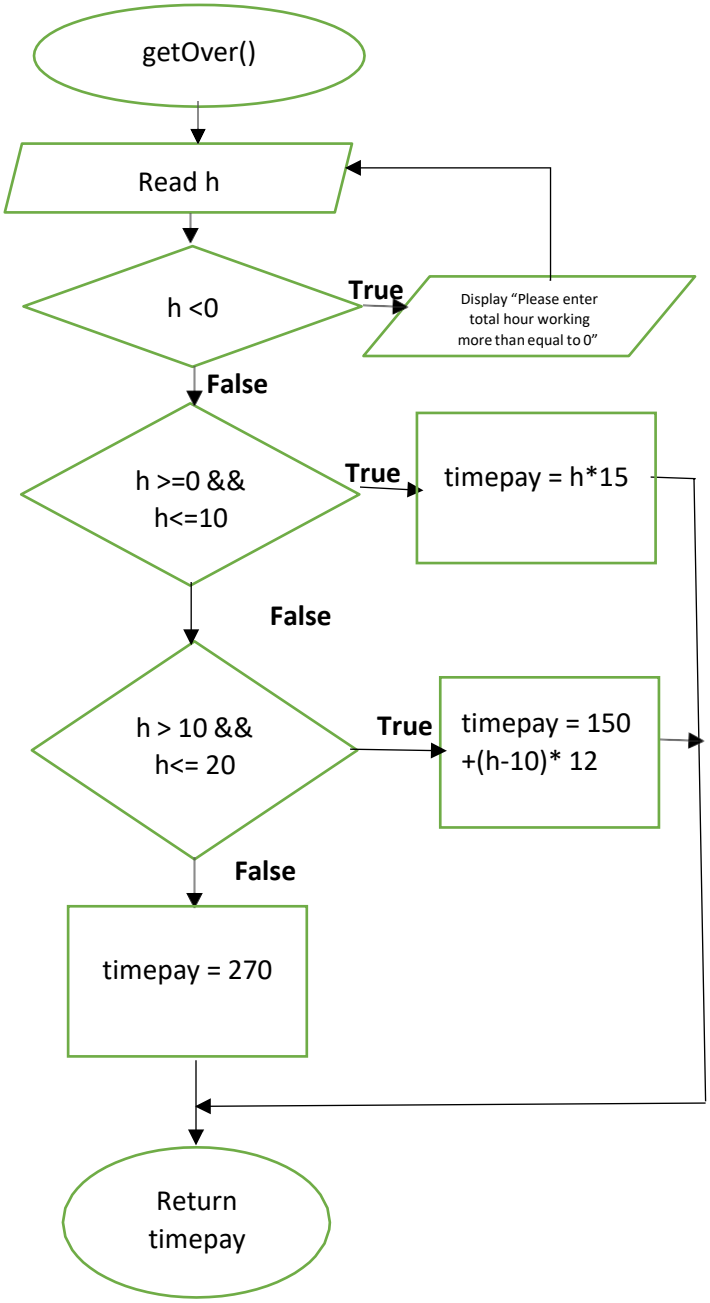

1
Read sizetoys
sizetoys != 'B'
&& sizetoys != 'S'
&& sizetoys != 'K'
True
Display
"Invalid Input
for toy's size"
False
Read amount
amount <0
True
Display "Please
enter amount more
than equal to 0"
False
A
sizetoys ==
'B'
True
net_salary = 8* amount
False
sizetoys ==
'S'
True
net_salary = 5* amount
False
net_salary = 2* amount
Display employee_name
,employee_number
,net_salary
END


getSalaryB()
Read masa
masa <0
True
Display “Please enter total hour working more than equal to 0”
False
masa <= 100
false
True
Salary = 100* 20
Salary = masa* 20
Return Salary
getSalaryS()
Read masa
masa < 0
True
Display “Please enter total hour working more than equal to 0”
False
masa >=0 && masa <= 50
True
Salary = masa * 10
False
masa <=100
False
True
Salary = 500 + 50*5
Salary = 500 + (masa -50) *5
Return Salary


getCode(
Read
c != 'G'
&&
c != 'K'
True
Display "Please choose G , K or S"
False
Return c


getStatus()
Read s
s!= 'P' &&
s!= 'B'
True
Display "Please choose either P or B"
False
Return s


getCategory()
Read k
k != 'B' &&
k != 'S'
True
Display "Please choose either B or S"
False
Return k


getOver()
Read h
h <0
True
Display "Please enter total hour working more than equal to 0"
False
h >=0 &&
h<=10
True
timepay = h*15
False
h > 10 &&
h<= 20
True
timepay = 150
+(h-10)* 12
False
timepay = 270
Return
timepay