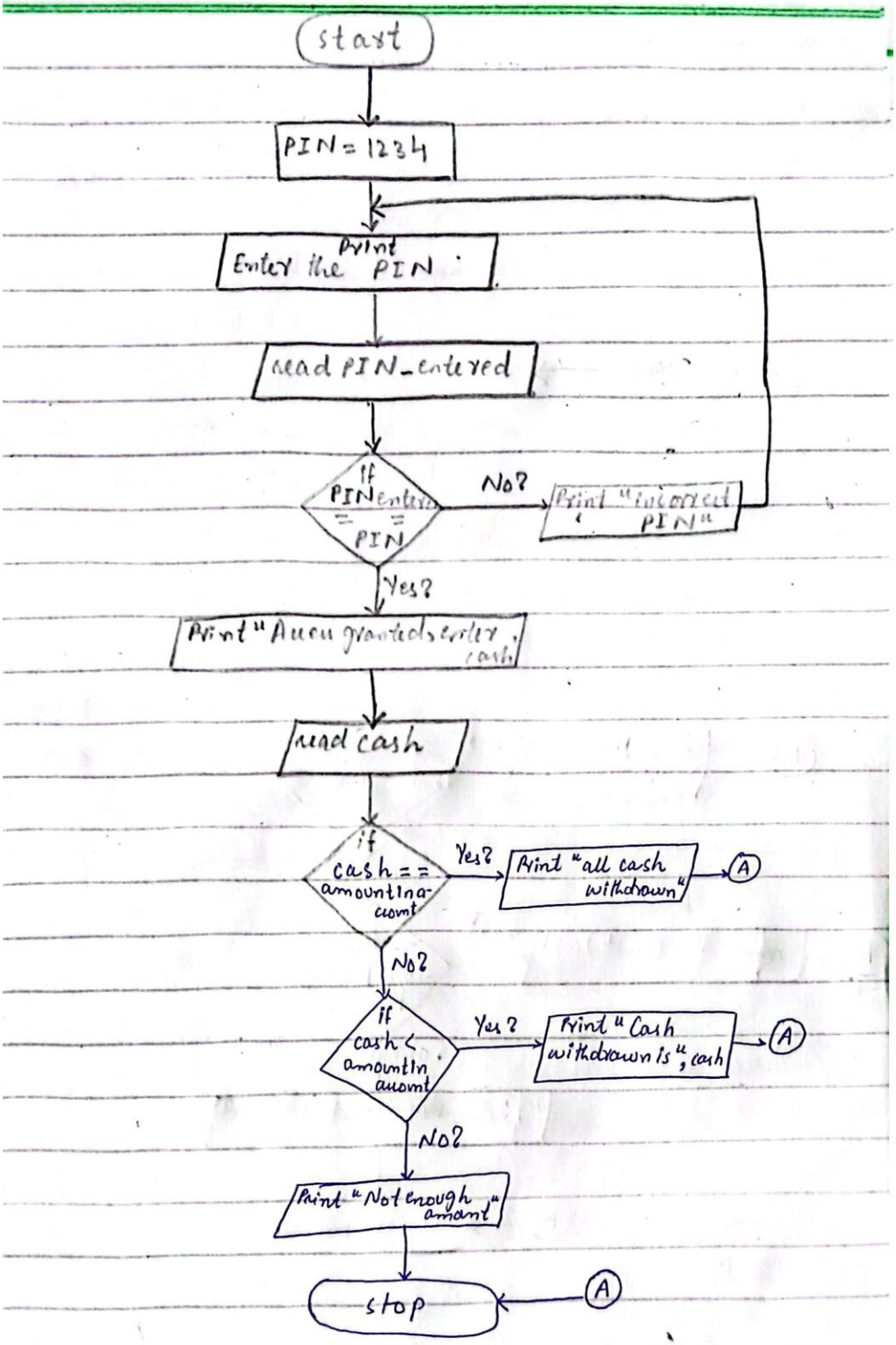
Question 2

Algorithm

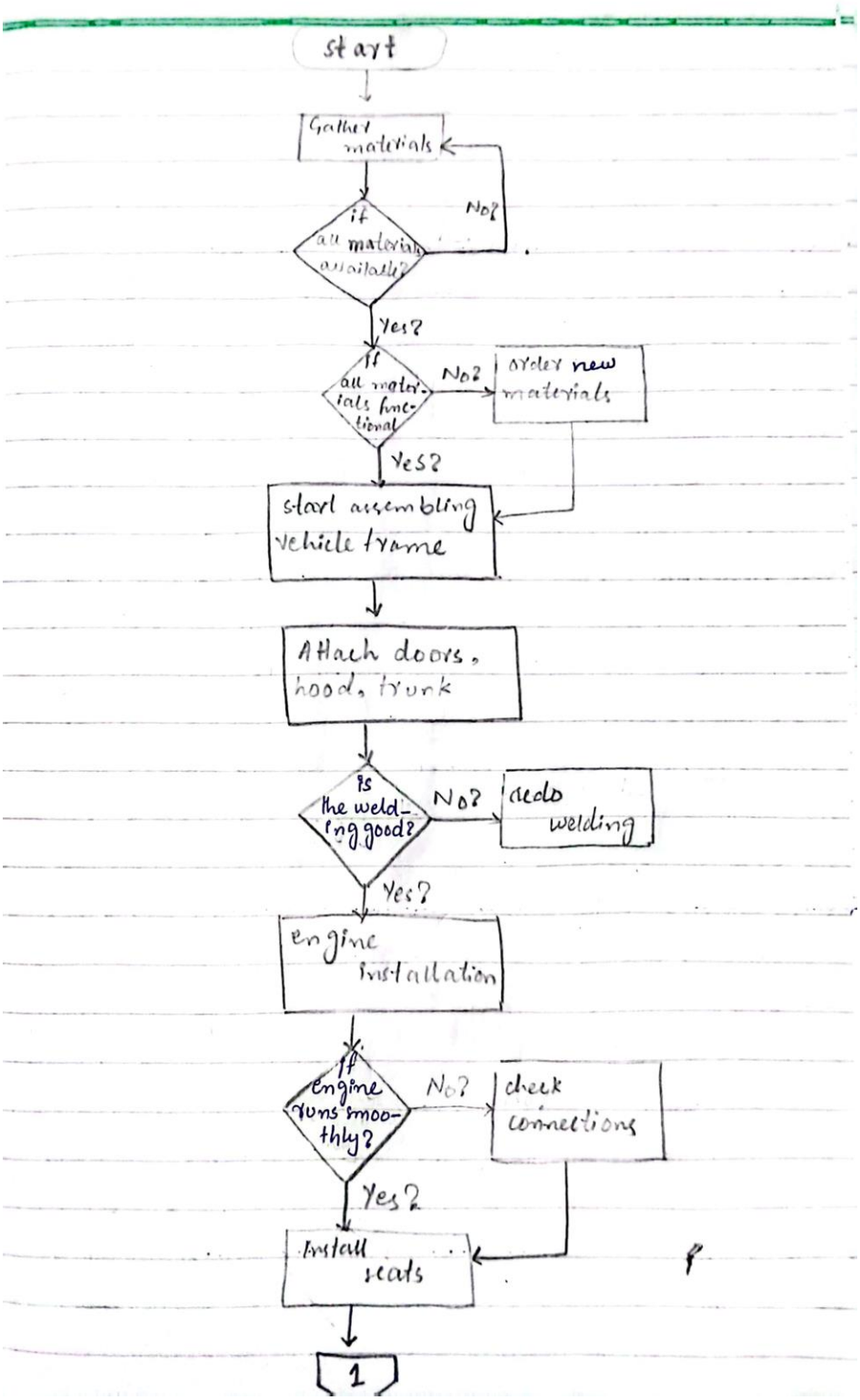
1. Start
2. Ask the user for PIN CODE
3. If CODE equals PIN then grant access
   ask for cash amount to withdraw
   check if enough cash is available
   else show warning
4. Else show incorrect PIN
5. Stop

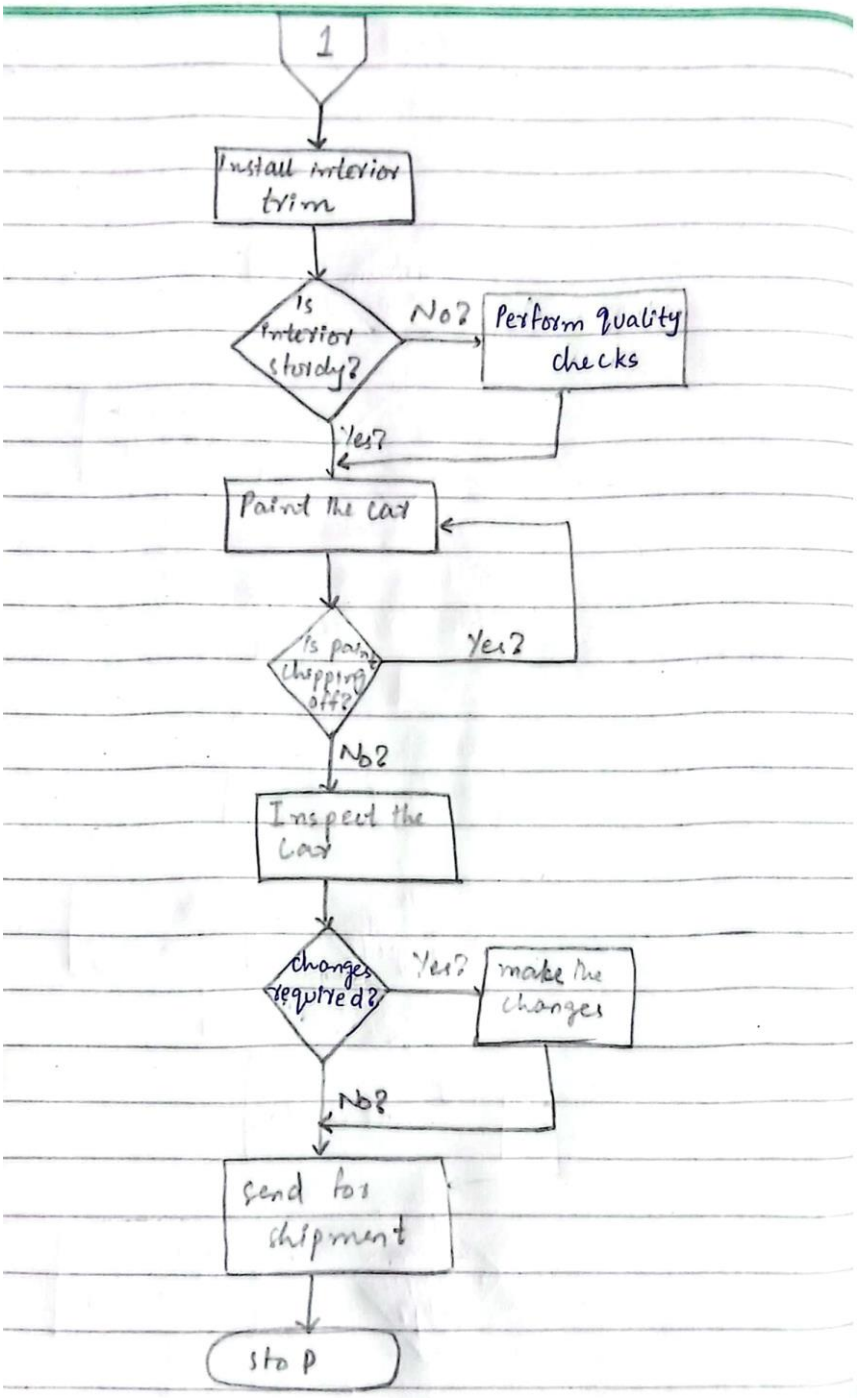
Pseudocode

1. Start
2. Set PIN=12345
3. Display “enter the PIN”
4. Read PIN_entered
5. If PIN_entered==PIN
   Then display “access granted, enter the cash amount”
   Read cash
   If cash==amountinaccount
   Then display “all cash withdrawn”
   Else if cash<amountinaccount
   Then display”cash withdrawn is”, cash
   Else
   Display”not enough cash in account”
6. Else
   Display”incorrect PIN”
   Goto step 3
7. Stop


start
PIN = 1234
Print Enter the PIN
read PIN_entered
if PIN entered == PIN
No?
Print "incorrect PIN"
Yes?
Print "Acct granted, enter cash"
read cash
if cash == amount in acount
Yes?
Print "all cash withdrawn"
A
No?
if cash < amount in acount
Yes?
Print "Cash withdrawn is", cash
A
No?
Print "Not enough amount"
stop
A

Question 6: Car assembly flowchart:




1
Install interior trim
Is interior sturdy?
No?
Perform quality checks
Yes?
Paint the car
Is paint chipping off?
Yes?
No?
Inspect the car
Changes required?
Yes?
make the changes
No?
send for shipment
stop

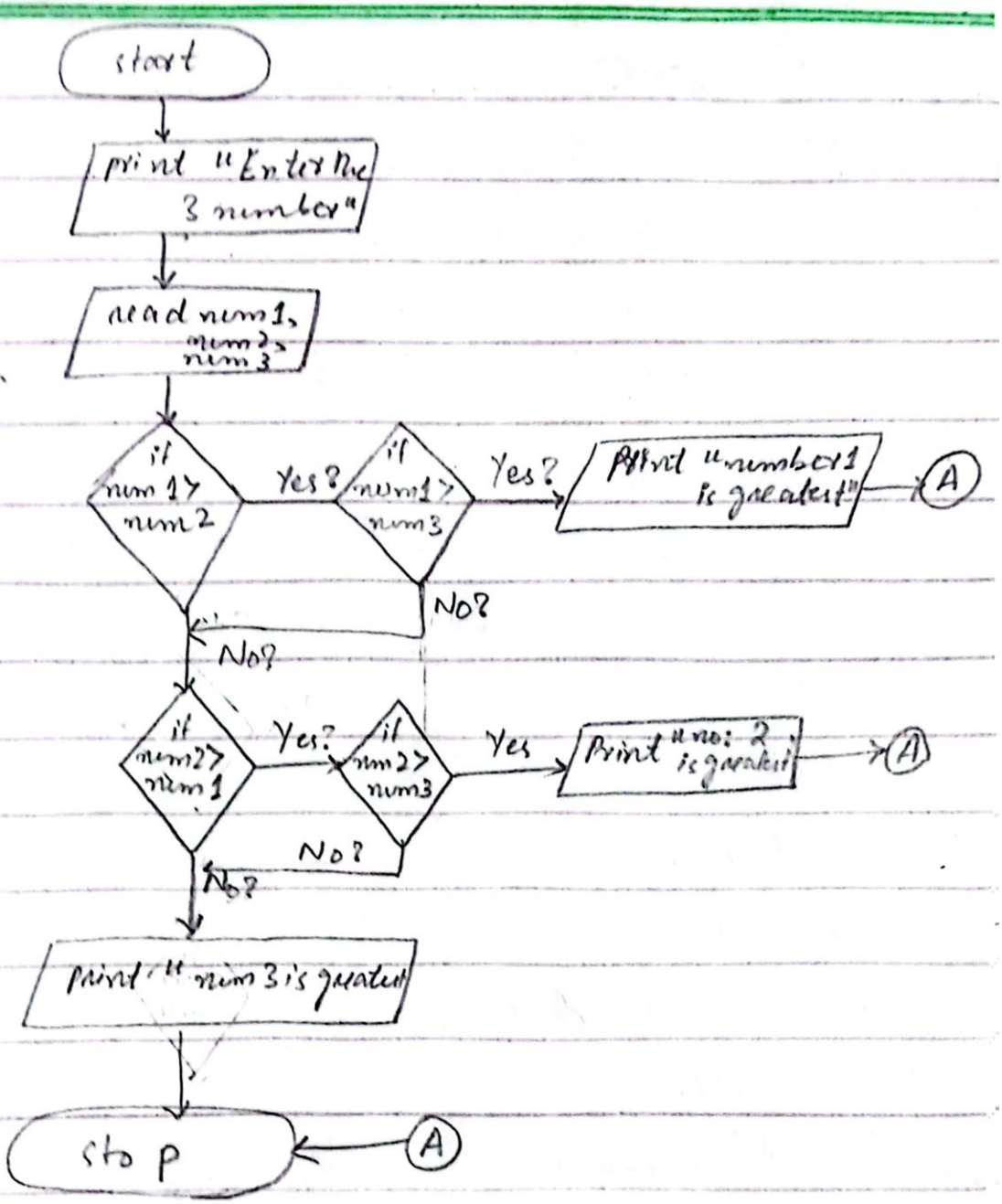
Question 3

Algorithm

1. Start
2. Ask the user for the first, second and third numbers
3. Take the numbers as input
4. If number 1>number2 and number1>number 3
   Then print number 1
5. Else if number 2> number3 and number 2> number 1
   Then print number 2
6. Else print number 3
7. Stop

Pseudocode

1. Start
2. Display “enter the 3 numbers”
3. Read num1, num2, num3
4. If num1>num2 && num1>num3
   Then display “the greatest number is:”, num1
   Else if num2>num3 && num2>num1
   Then display “the greatest number is”, num2
   Else
   Display”the greatest number is:”, num3
5. Stop


start
print "Enter the 3 number"
read num1, num2, num3
if num1 > num2
Yes?
if num1 > num3
Yes?
Print "number1 is greatest"
A
No?
No?
if num2 > num1
Yes?
if num2 > num3
Yes
Print "no: 2 is greatest"
A
No?
No?
Print "num3 is greatest"
stop
A

Question 1

Algorithm

1. Start
2. Greet the customer
3. Show the customer the menu
4. Ask the customer their order
5. Tell the customer about the availability of the item
6. Ask the customer about add ons
7. Calculate the bill
8. Ask the customer for cash
9. Check if change needs to be given or cash is insufficient
10. Thank the customer
11. End

Pseudocode

1. Start
2. Display “welcome to our restaurant”
3. Display “here’s the menu”
4. Read order
5. If order==”available”
   Then display “your order has been placed”
   Else
   Display “item unavailable please order again”
   Goto step 4
6. Display “any add ons you’d like to have?”
7. Read answer

8. If answer==”yes”

    Display “what add ons would you like?”

    Read add_on

   Else

    Display “your order is done”

9. Calculate bill
10. Get cash
11. If cash==bill

    Then display “thankyou, your order will shortly be given to you”

    Else if cash>bill

    Then change=cash-bill

    Display”here’s your change of rupees”,change

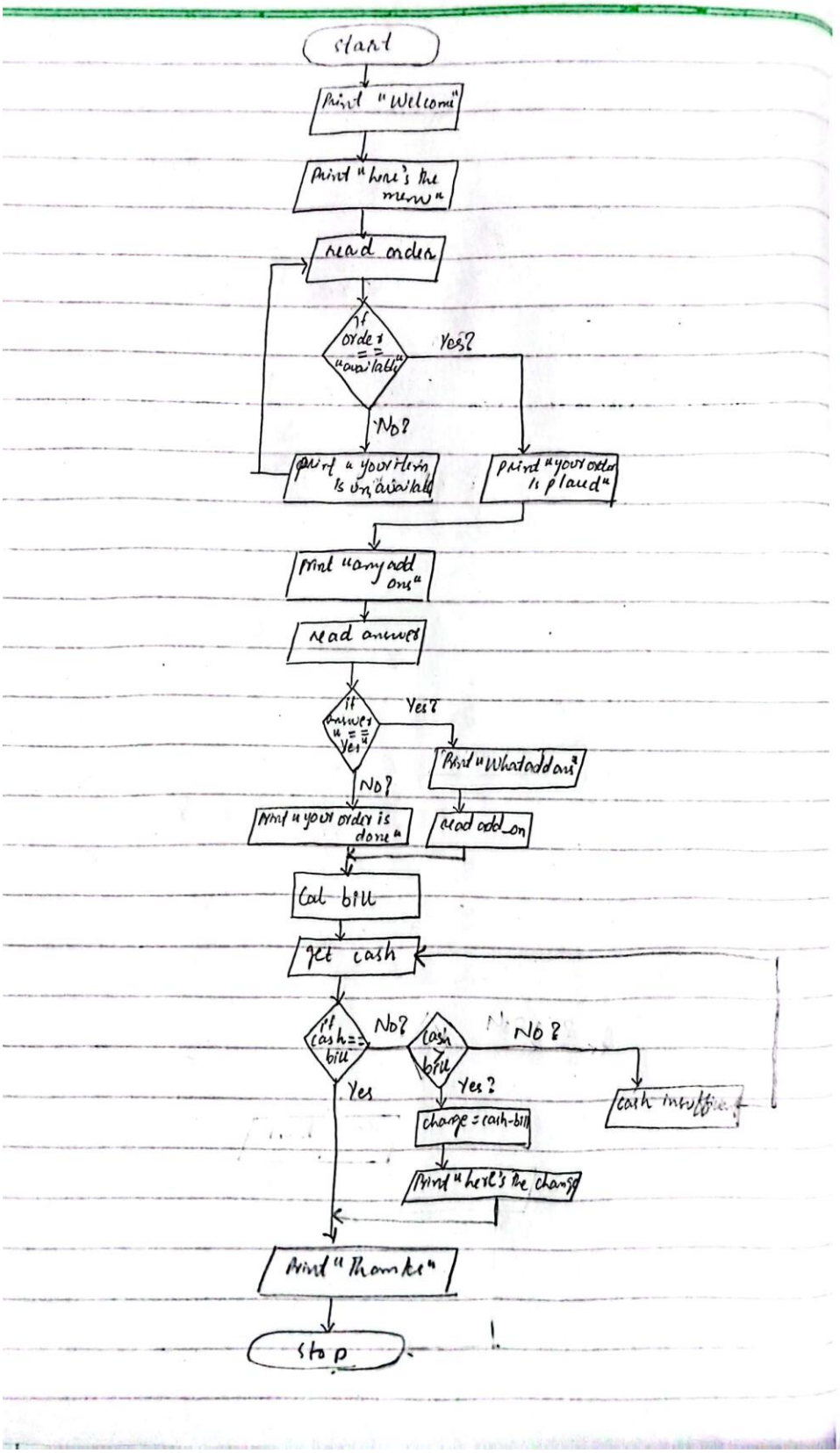
    Else

    Display”cash insufficient”

    Goto step 10

    End if

12. End


start
Print "Welcome"
Print "here's the menu"
read order
if order == "available"
Yes?
No?
print "your item is unavailable"
print "your order is placed"
Print "any add ons"
read answer
if answer == "Yes"
Yes?
Print "What add on?"
No?
Print "your order is done"
read add_on
Cal bill
get cash
if cash == bill
No?
cash > bill
No?
Yes
Yes?
change = cash-bill
Print "here's the change"
Print "Thanks"
stop

Question 4

ALGORITHM

1. Start
2. Read number from user
3. Check If number>=1 AND number<=12
   If yes then proceed to the next step
   If no then print invalid number
4. Check if number==1
   Print JANUARY
5. Else check if number==2
   Print FEBRUARY
6. Else check if number==3
   Print march
7. Else check if number==4
   Print APRIL
8. Else check if number==5
   Print MAY
9. Else check if number==6
   Print June
10. Else check if number==7
    Print July
11. Else check if number==8
    Print august
12. Else check if number==9
    Print September
13. Else check if number==10
    Print October
14. Else check if number==11

Print November

15. Else print December
16. stop

Question 5

PSEUDOCODE

1. START
2. Read num1, num2, operator
3. If operator==’+’

        Then sum=num1+num2

            Display “the answer is”, sum

    Else

        Subtract=num1-num2

        Display ”the answer is” , subtract

4. stop

Question 7

ALGORITHM

1. START
2. Read num1, num2, operator from the user
3. Check If operator is equal to ’+’
   Then sum=num1+num2
   Print “the sum is”, sum
4. Else check if operator is equal to ’-‘
   Then subtract=num1-num2
   Print”the answer is”, subtract
5. Else check if operator is equal to ’/’
   Then check if num2 is equal to 0
   Print not divisible by 0
   Else
   Div=num1/num2
   Print “the answer is”, div
6. Else check if operator is equal to ’*’
   Then product=num1*num2
   Print “the answer is”, product
7. Else Print “invalid operator”
8. stop

Question no:9

Why we use .gitignore?

Answer:

We use ".gitignore" files to tell git what files to ignore when tracking changes

- First it avoids clutter by ignoring temporary files which are not essential for version control. There are also certain files with some private information such as files with API keys and .gitignore ensures that these aren't accidentally committed.
- It also improves the performance of git by increasing it's speed.
- Using .gitignore file as a text file, that specifies git which files to ignore, also reduces the size of your repository.

Question 10:

Difference between an algorithm an pseudocode.

Answer:

Algorithm:

An algorithm is a step by step instructions process to solve a problem/task. Algorithms are described in natural language and provide a high level view of the process. They aren't tied to any specific programming language or syntax. An algorithm is abstract, focusing on general ideas.

Pseudocode:

Pseudocode can be explained as the simplified version of programming code which isn't specific to one particular language. It uses structured syntax and control structures. It is more specific than an algorithm as it uses actual code like constructs and notations. It looks more similar to actual code.