
# ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY

## COLLEGE OF NATURAL AND SOCIAL SCIENCE

## Department of software engineering

### Fundamentals of programing -Group project


| NAME | ID. |
|---|---|
| 1. SAMUEL GIRMA | ETS1421/14 |
| 2.SOLOMON TADESSE | ETS1495/14 |
| 3.SURAFEL ABERA | ETS1507/14 |
| 4.YONAS GEMECHU | ETS1733/14 |
| 5.YORDANOS LEGESSE | ETS1760/14 |
| 6.ZEKARIYAS KUMSA | ETS1789/14 |


**Group-5**

**SECTION : E**

**SUBMITTED TO: ELENI TESHOME**

**SUBMISSION DATE: April 8, 2023**

# *Problem analysis-Part I*

## *Problem:*

It is difficult to make a budget that spans several years, because prices are not stable. If your company needs 200 pencils per year, you cannot simply use this year's price as the cost of pencils 2 years from now. Because of inflation the cost is likely to be higher than it is today. Write a program to gauge the expected cost of an item in a specified number of years. The program asks for the cost of the item, the number of years from now that the item will be purchased, and the rate of inflation as a percentage, like 5.6 (percent). Your program should then convert the percent to a fraction, like 0.056, and should use a loop to estimate the price adjusted for inflation. The program then outputs the estimated cost of the item after the specified period. The program should also display the cost difference between every year in tabular until the specified period. But allow the user to choose which price of the pencil he/she want to see (either the pencil cost of specific year from now on wards or each year of specified number of years in tabular format.)

## *Goal:*

- Solve the problem by using appropriate method
- Write a program that sets an estimated budget from given input

## Input:

- Budget starting and ending year
- Current price of each pencil(ETB)
- Quantity of pencil
- Estimated inflation rate (%)

## *Output:*

- The estimated cost of the item after the specified period
- The cost difference between every year in tabular format either for a specific year or for each year of the specified period.

# *Problem design*

## *I. Pseudocode:*

Step1: start

Step2: read permission

Step3: if(permission==1)then

Step4: read –: starting year,ending year,each  item price, starting inflation rate, numberof pencil

Step5:rateofinflation=starting inflationrate

5.1 Totalprice= number of pencil*each item price

5.2 budget year=ending year-starting year

Step6:declar array:estimatedprice[budgetyear]

Step7:i=0

Step8: estimatedprice[i]=totral price +rateofinflation

Step9:rateofinflation=rateofinflation*2

Step10:i=i+1

Step11:if(i<=budgetyear)then go to step 8

Step12:Ask the user choice to see all budget or specific year budget

Step13read choice

Step14:if(choice==1)then

14.1:i=0

14.2:print estimatedprice[i]

14.3:i=i+1

14.4:if(i<=budgetyear)then go tostep14.2

Step15:if(choice==2)then read choiceyear

Step16:checkyear=startingyear

Step17:rateofinflation=startinginfationrate

Step18:i=0

18.1:if(choiceyear==checkyear)then

18.1.1:print estimatedprice[i];

break;

18.2:each itemprice=eachitemprice+rateofinflation

18.3:rateofinflation=rateofinflation*2

18.4:check year=checkyear+1

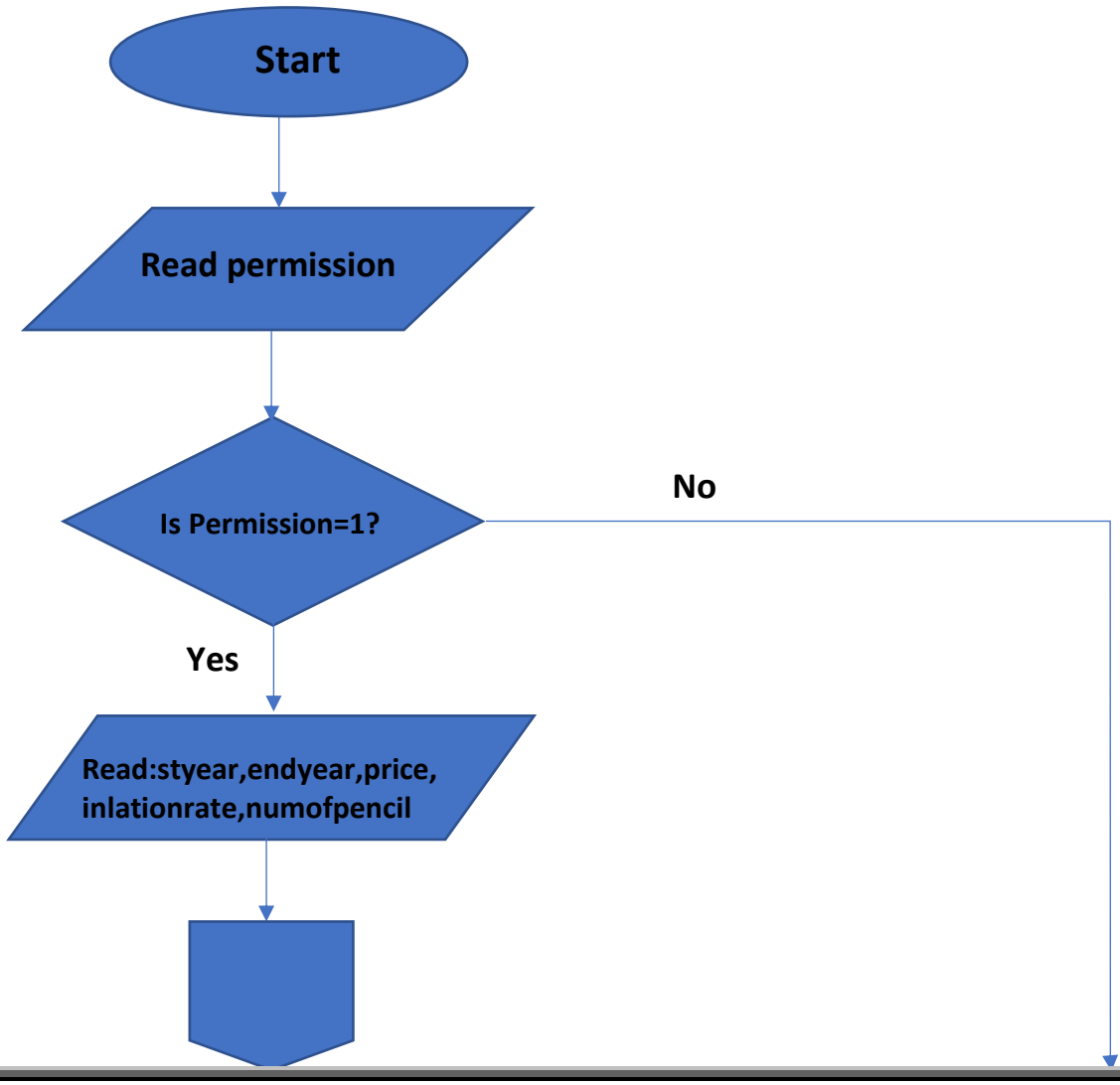
18.5:if(i<=budgetyear)then goto step18.1

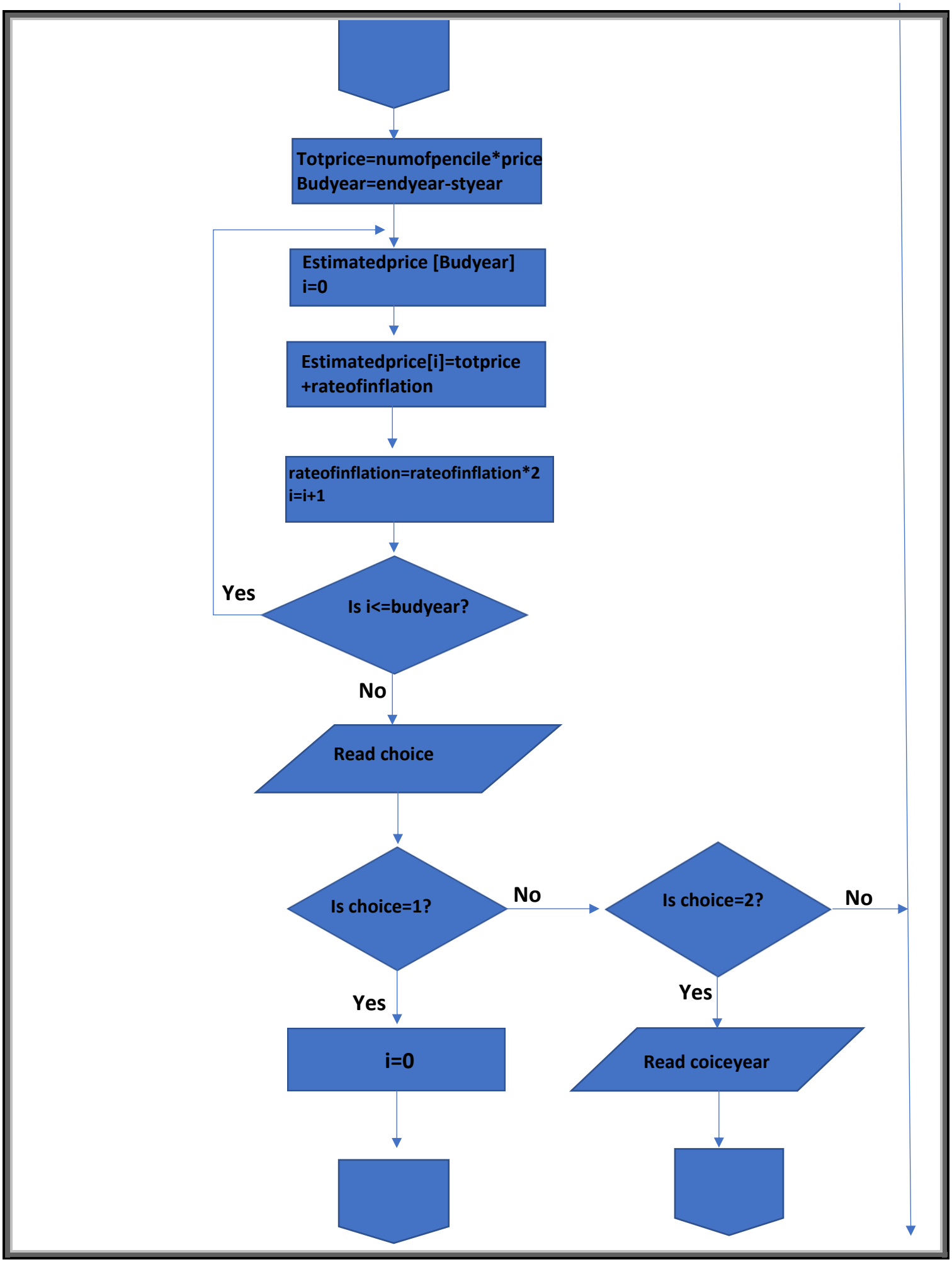
Step19:if(choice!=1&&choice!=2) then exit the program

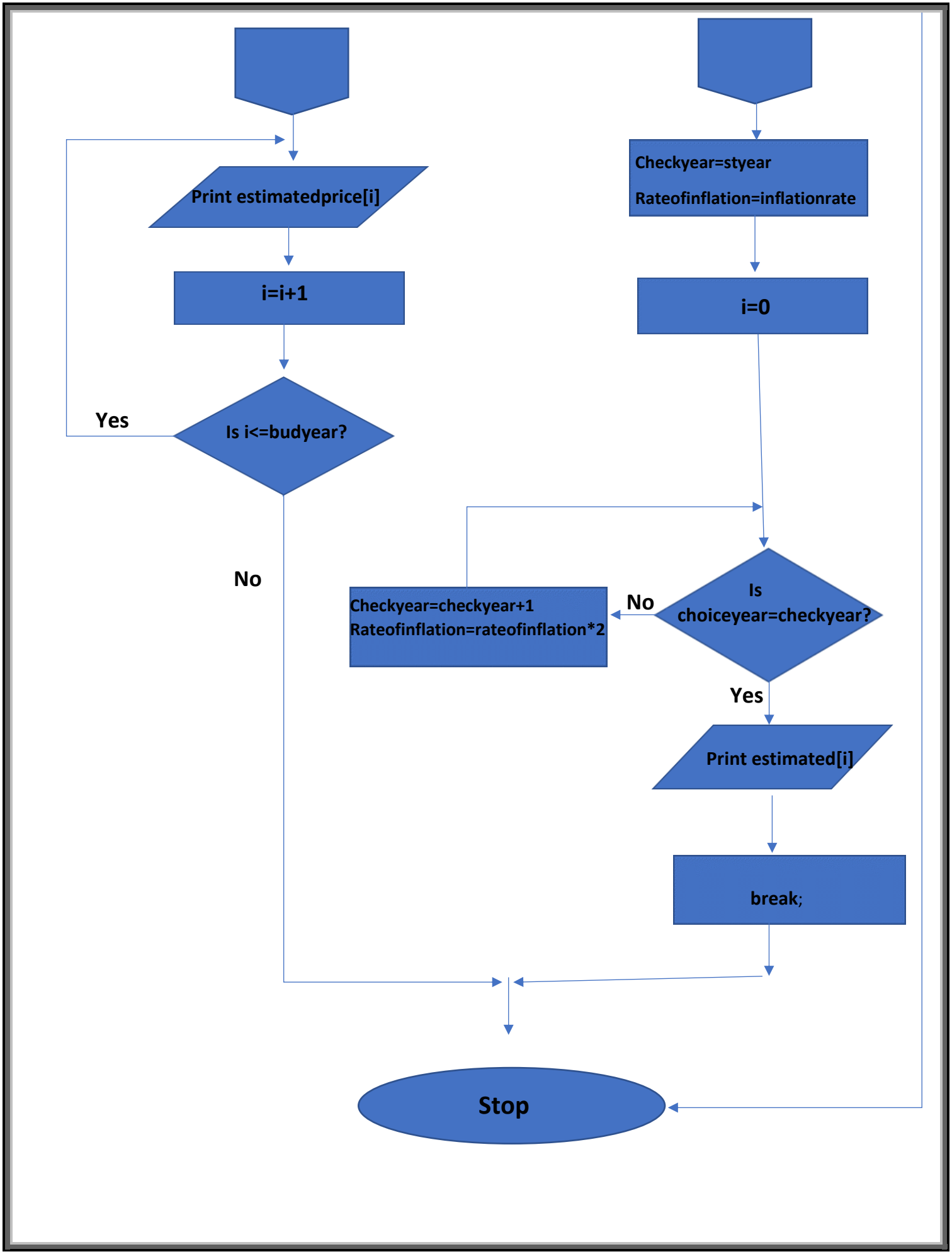
Step20:if(permission!=1) then exit the program

Step 21: Stop

## Flowchart




Totprice=numofpencile*price
Budyear=endyear-styear
Estimatedprice [Budyear]
i=0
Estimatedprice[i]=totprice
+rateofinflation
rateofinflation=rateofinflation*2
i=i+1
Yes
Is i<=budyear?
No
Read choice
Is choice=1?
No
Is choice=2?
No
Yes
Yes
i=0
Read coiceyear


Print estimatedprice[i]
i=i+1
Is i<=budyear?
Yes
No
Checkyear=styear
Rateofinflation=inflationrate
i=0
Checkyear=checkyear+1
Rateofinflation=rateofinflation*2
No
Is choiceyear=checkyear?
Yes
Print estimated[i]
break;
Stop

## *Part II-Pattern b and c*

### *Pseudocode-b*

**Step1: start**

**Step2: Display the message "enter the number n and -1 to quit the program"**

**Step3: read n**

**Step4: If (n< = 0  and n != -1), then goto step 20**

**Step5:  i = n**

**5.1:  If (i < n), then x=n;**

**5.2: print space;**

**5.3 x--;**

**5.4if x>i then goto step 5.2**

**Step6: j=0**

**Step7: print j**

**7.1: if (j==i) then k=i-1**

**7.2print k**

**7.3 k--;**

**7.3 if (k>=0) then goto step: 7.2**

**Step8: j++**

**Step9: if (j<=i) then got Step7**

**Step10: i--**

**Step11: if (i>=0) then go to step 5.1**

**//End of the upper triangle**

**Step12:  i = 1**

**12.1:  If (i < n), then x=n;**

**12.2: print space;**

**12.3 x--;**

**12.4if x>i then goto step 12.2**

**Step13: j=0**

**Step14: print j**

**14.1: if (j==i) then k=i-1**

**14.2print k**

**14.3 k--;**

**14.3 if (k>=0) then goto step: 14.2**

**Step15: j++**

**Step16: if (j<=i) then got Step 14**

**Step17: i++**

**Step18: if (i<=n) then goto step 12.1**

**step19: if (n <= 0 and n !=-1 )**

**step20: Display you entered n and goto step 2**

**step 21: if( n!= -1)**

**Step 22: Display do you want to try again**

**Step 23: read input_validation**

**Step 24: if ( input_validation == y or y)**

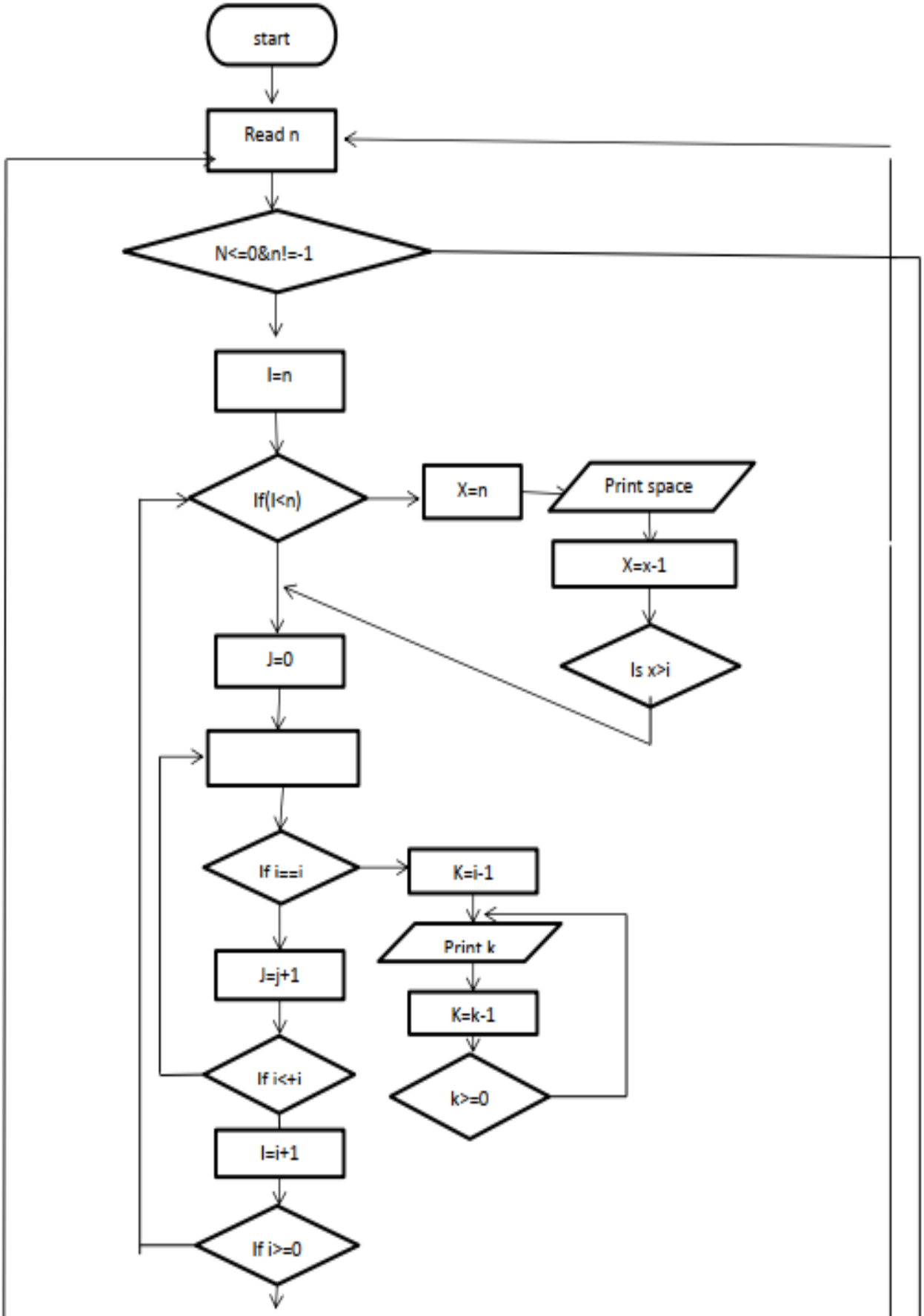
**Step 25: go to step 2**

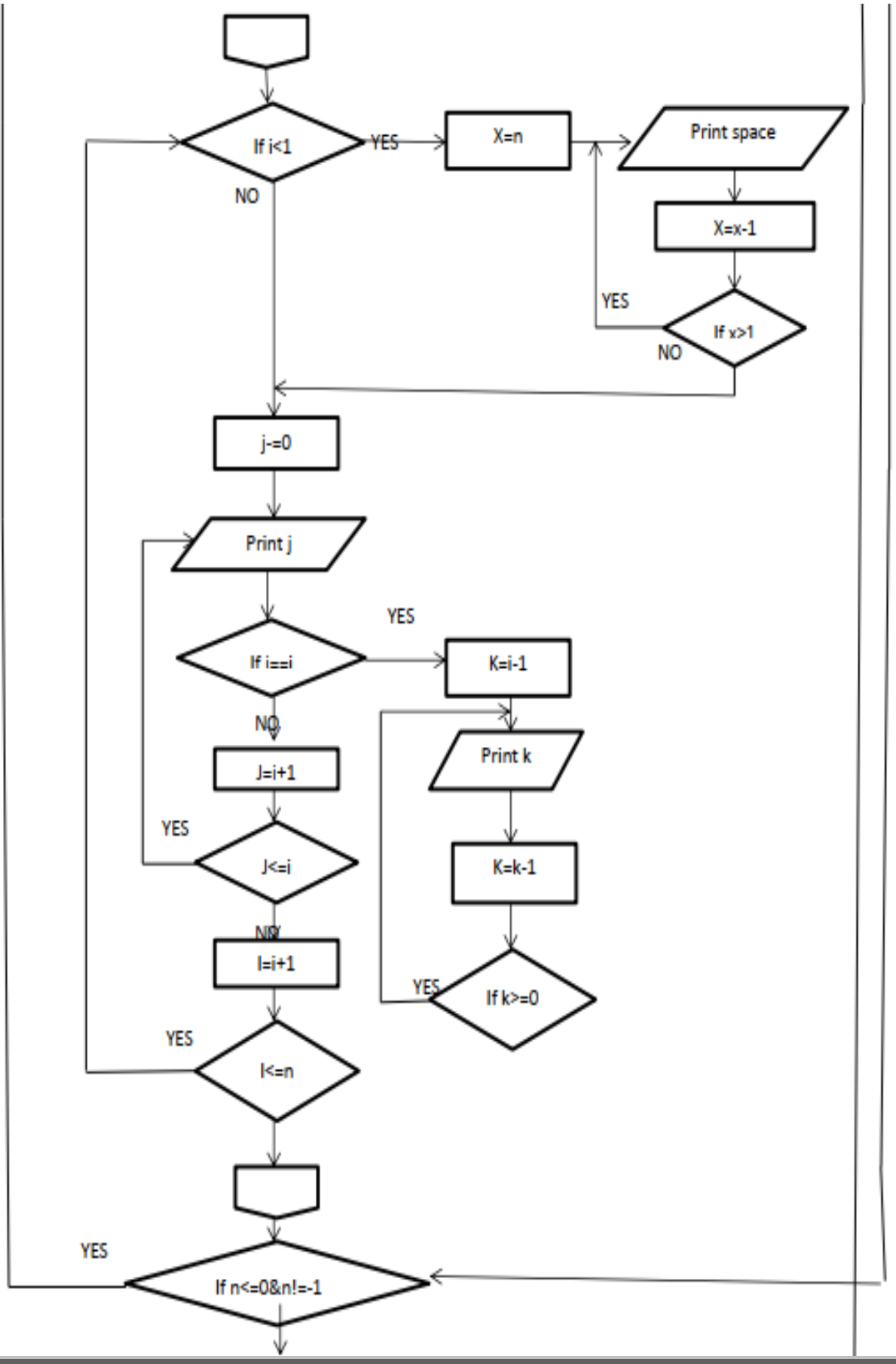
**Step 26 else**

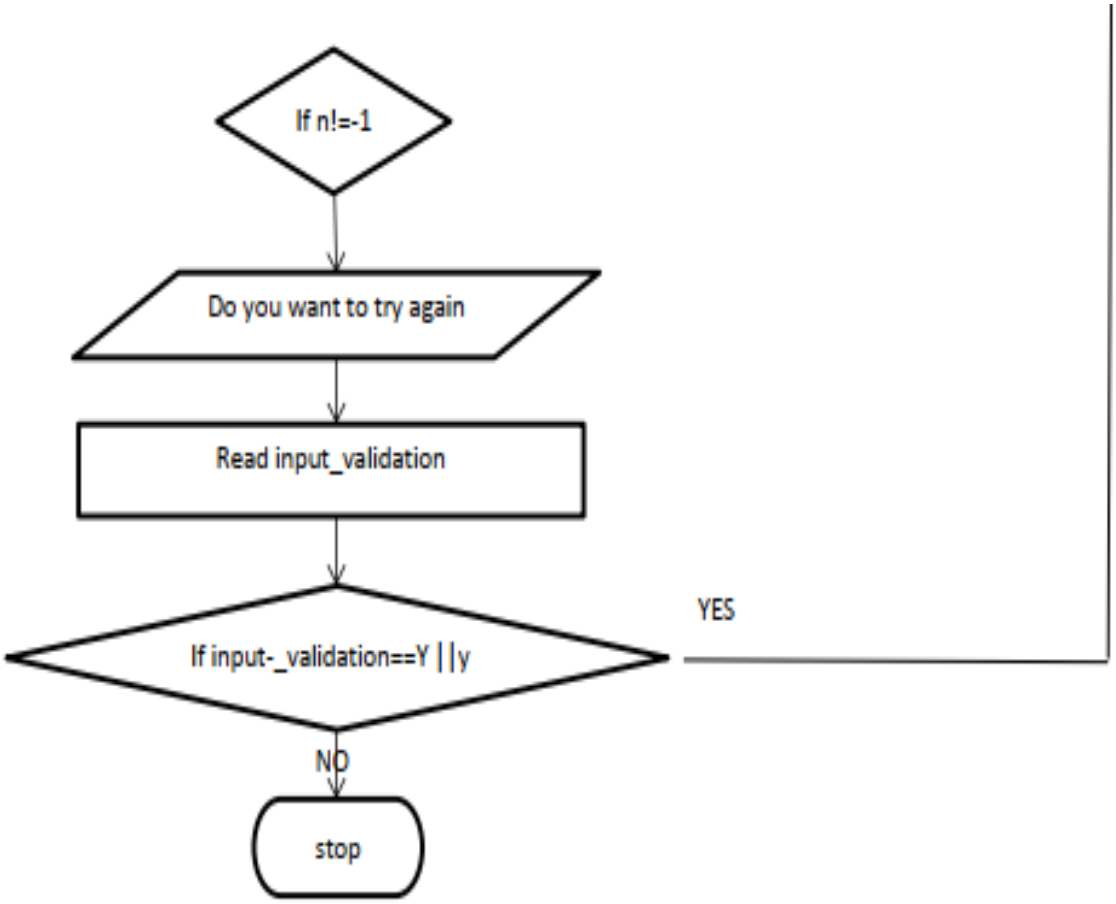
**Step 27 stop**

**//End of the lower triangle**

## Flowchart-b


start
Read n
N<=0&n!=-1
I=n
If(I<n)
X=n
Print space
X=x-1
Is x>i
J=0
If i==i
K=i-1
Print k
K=k-1
k>=0
J=j+1
If i<+i
I=i+1
If i>=0


If i<1
YES
X=n
Print space
NO
X=x-1
YES
If x>1
NO
j-=0
Print j
YES
If i==i
K=i-1
NO
J=i+1
Print k
YES
J<=i
K=k-1
NO
I=i+1
YES
If k>=0
YES
I<=n
YES
If n<=0&n!=-1




## *Pseudocode-c*

**Step1: start**

**Step2: Display the message "enter the number which is greater than 0 and -1 to quite the program"**

**Step3: read n**

**Step4: if n!=0 and n!=1 then**

**Step5: i = 0**

**Step6: j = i**

**Step6.1: print j**

**Step6.2: if ( i != n) then**

**Step6.21: if (j==0)**

**step6.211: space = 0**

**step6.212: print space (blank space)**

**step6.213: space ++**

**step6.214: if space <= 2*(n-i)-2, then go to step 6.212**

**step6.215: k=0**

**step6.216: print k**

**step6.217: k++**

**step6.218: if K<=i, then goto step 6.216**

**step6.3: if j==0**

**step6.31: if i==n**

**step6.311: k=1**

**step6.312: print K**

**step6.313: k++**

**step6.314: if k<=i, then goto step 6.312**

**step6.4: j- -**

**step6.5: if j>=0 then go to 6.1**

**step7: i++**

**step8: if i<=n then goto step 6**

**step9: i=n-1**

**step10: j=i**

**step10.1: print j**

**step10.2: if j==0**

**step10.21: space=0**

**step10.22:print space (blank space)**

**step10.23:space++**

**step10.24: if space <=2*(n-i)-2 then goto step 10.22**

**step10.25: k=0**

**step10.26: print k**

**step10.27: k++**

**step0.28: if k<=i then goto step 10.26**

**step10.3: j - -**

**step10.4: if j >=0 , then goto 10.1**

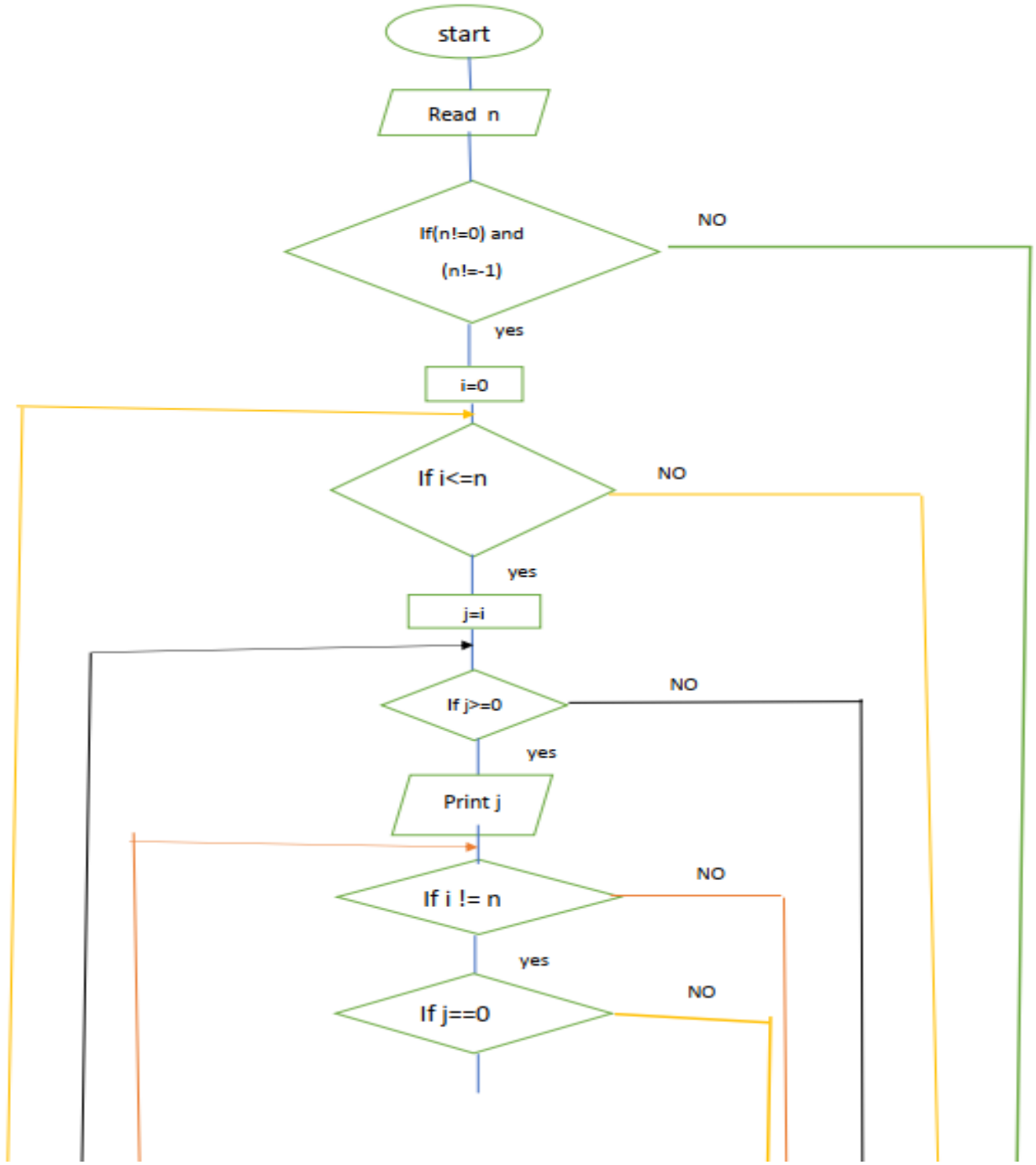
**step11: i - -**

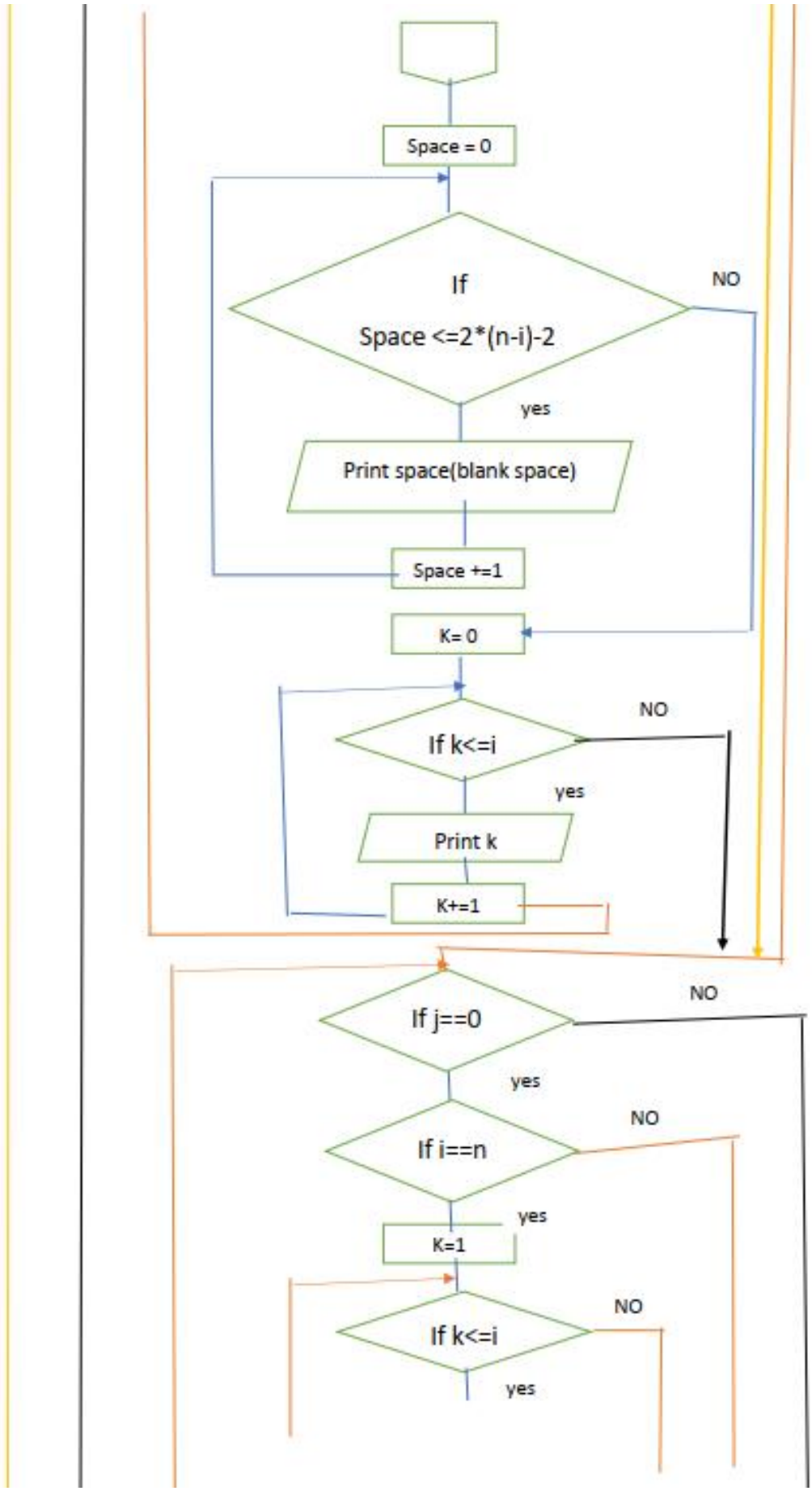
**step12: if i>=0, then goto step 10**

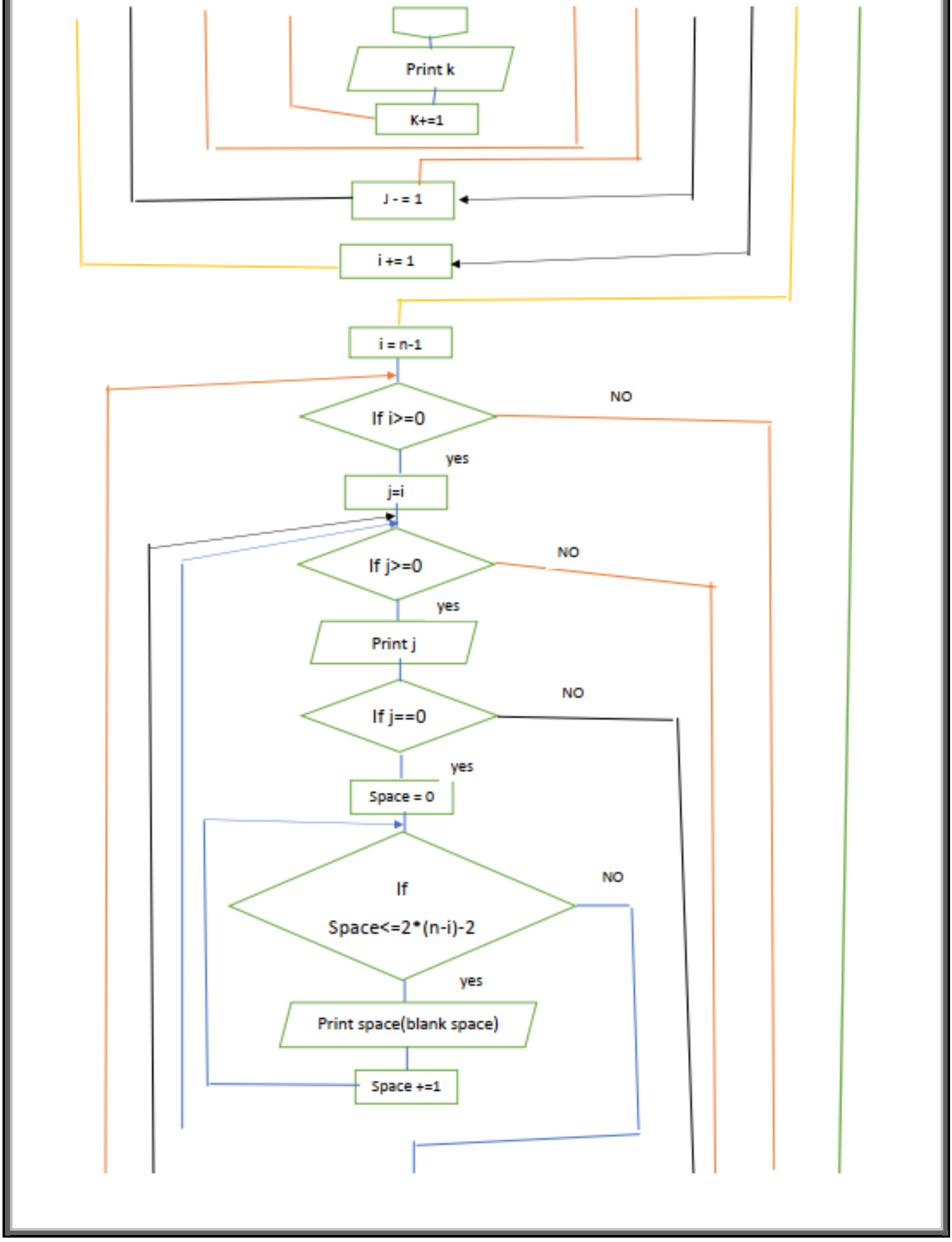
**step13:else if n==0 or n== -1 , then exit**

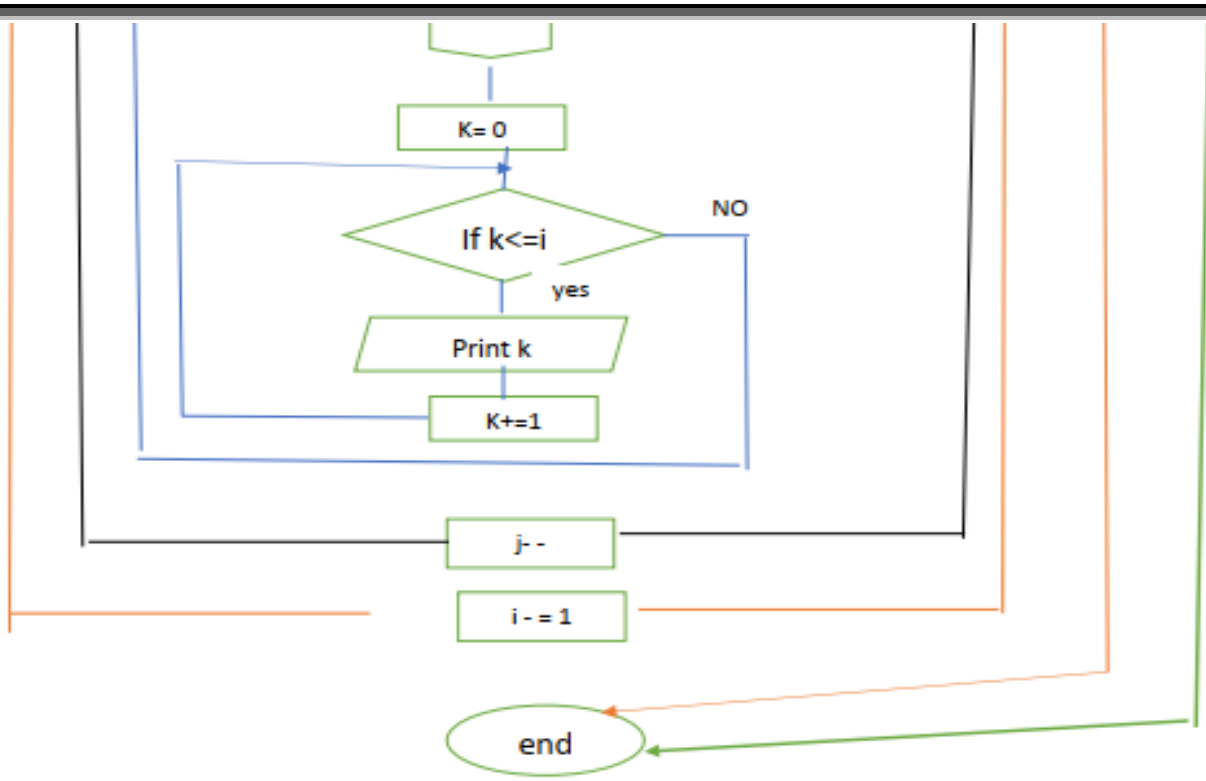
**step14: stop**

## Flowchart-c




Space = 0
If
Space <=2*(n-i)-2
NO
yes
Print space(blank space)
Space +=1
K= 0
If k<=i
NO
yes
Print k
K+=1
If j==0
NO
yes
If i==n
NO
yes
K=1
If k<=i
NO
yes


Print k
K+=1
J - = 1
i += 1
i = n-1
If i>=0
NO
yes
j=i
If j>=0
NO
yes
Print j
If j==0
NO
yes
Space = 0
If
Space<=2*(n-i)-2
NO
yes
Print space(blank space)
Space +=1


K= 0
If k<=i
NO
yes
Print k
K+=1
j- -
i - = 1
end