1. A bear starting from the point P, walked one mile due south, Then he changed direction and walked one mile due east. Then he turned again to the left and walked one mile due north and arrived at point P he started from what was the color of the bear?

Algorithm:

i)Start at Point P.

ii)Move one mile south.

iii)Move one mile east.

iv)Move one mile north.

v)Check if the bear returns to the starting point P.

vi)If the bear returns to Point P, check if P is near the North Pole.

vii)The North Pole is the only general location where this is possible.

viii)There are special cases near the South Pole where this condition is met, but bears don’t live there.

ix)If P is the North Pole, the only bear found there is the polar bear.

x)Polar bears are white in color.

xi)Output "White" as the bear’s color.

2.Two towns A and B are 3 km s apart It is proposed to build a new school serving 100 students in town A and 50 students in town B. How far from town A should the school be built if the total travel distance by all 150 students is to be as small as possible?

Algorithm:

i) Define two towns, A and B, separated by 3 km.  
ii) Identify the number of students in each town: 100 students in Town A and 50 students in Town B.  
iii) The school should be placed so that the distances from A and B are in the inverse ratio of the number of students.  
iv) Since Town A has 100 students and Town B has 50 students, the distance should be divided in a 1:2 ratio (since 100:50 simplifies to 2:1).  
v) Divide the 3 km distance in a 2:1 ratio:

Distance from Town A = (2/3) × 3 km = 2 km

Distance from Town B = (1/3) × 3 km = 1 km  
vi) The school should be built 1 km from Town A and 2 km from Town B.  
vii) Print the result:

"The school should be built 1 km from Town A."

"Or 2 km from Town B for minimal travel distance."  
viii) End the process.

3.A traveller arrives at hotel he has no money but only a silver chain consisting of 6 links. He uses one link to pay for each day spent at the hotel but the hotel manager agrees to accept no more than one broken link.How should the traveller cut up the chain in order to settle the

amount with the hotel manager on a daily basis

I)what is the least number of links that have to be cut if the traveller stays 100 days at the hotel and has a chain cosisting

of 100 links? what is the answer in general case n days and n links

Algorithm:

Understanding the Problem

i)The traveler has a silver chain with N links.

ii)He must pay one link per day to the hotel manager.

iii)The manager allows only one broken link at any time.

iv)The goal is to determine the minimum number of cuts required to make daily payments for N days.

Case 1: 6-Link Chain for 6 Days

The optimal strategy follows the binary (power of 2) system:

i)Break the chain strategically so that different combinations allow for daily payments.

ii)Cut at 1st, 2nd, and 4th links to create pieces of 1, 2, and 3 links.

iii)Payments can be made as follows:

Day 1: Give 1-link piece.

Day 2: Take back the 1-link piece and give the 2-link piece.

Day 3: Give both 1-link and 2-link pieces (total 3).

Day 4: Take back 1-link and 2-link pieces and give the 4-link piece.

Day 5: Give the 1-link piece along with the 4-link piece.

Day 6: Give the 2-link piece along with the 4-link piece.

iv)Minimum cuts needed = 2 (at positions 1 and 3).

Case 2: 100-Link Chain for 100 Days

i)To generalize, we use the binary representation of numbers to minimize cuts.

ii)The best way to split a chain is into pieces of sizes powers of 2:

Example: 1,2,4,8,16,32,37(since 1+2+4+8+16+32+37=100).

iii)The number of cuts required is the number of pieces needed:

For 100 links, we need 6 cuts at strategic positions.

iv)Thus, minimum cuts for 100 days = 6.

General Case: N Days & N Links

i)For N links, we minimize the number of cuts by dividing into powers of 2:

The number of cuts required is log₂(N) (rounded up to the nearest integer).

ii)This ensures that we always have the correct amount for each day.

1. Rearrange the letters in the words new door to make one word

Algorithm:

i)Input: Take the phrase "new door".

ii)Remove spaces to consider only "newdoor".

iii)Convert all characters to lowercase (for case insensitivity).

iv)Use a permutation function to generate all possible rearrangements of "newdoor".

v)Compare each permutation with "oneword".

vi)If a match is found, store the valid rearrangement.

vii)If "one word" is found, print "Rearranged successfully".

viii)Otherwise, print "No valid rearrangement found".

ix)End the process.

1. Do divide and conquer 6 5 1 4 3 2.

Algorithm:

i)Divide the Array:

If the array has more than one element, split it into two halves:

Left half = first n/2 elements.

Right half = last n/2 elements.

ii)Recursively Sort Each Half:

Apply Merge Sort recursively on both halves until each subarray contains only one element.

iii)Merge the Sorted Halves:

Compare elements from both halves and merge them in sorted order.

iv)Merge Sort Applied to [6, 5, 1, 4, 3, 2]:

Step 1 (Divide): [6, 5, 1] and [4, 3, 2]

Step 2 (Divide Again): [6, 5] and [1], [4, 3] and [2]

Step 3 (Divide Further): [6] and [5], [4] and [3]

v)Merging Steps:

[6] and [5] → [5, 6]

[4] and [3] → [3, 4]

Merge [5, 6] with [1] → [1, 5, 6]

Merge [3, 4] with [2] → [2, 3, 4]

Final Merge: [1, 5, 6] and [2, 3, 4] → [1, 2, 3, 4, 5, 6]

vi)Final Output: [1, 2, 3, 4, 5, 6]

6.Draw flowchart for calculating simple interest

Flowchart:

i)Start

ii) Input Principal (P), Rate of Interest (R), and Time (T)

iii)Calculate Simple Interest using the formula = (P\*R\*T)/100

iv)Display the Simple Interest

v)End

**Calculate**

**SI=(P\*R\*T)/100**

**Start**

**Input: P,R,T**

**End**

**Display SI**