**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?**

Ans. Algorithm

1. Start at point P.
2. Move one mile south.
3. Move one mile east.
4. Move one mile north.
5. Check if the bear returns to the starting point P.
6. If P is the North Pole, this condition holds true.
7. The only bears found in the North Pole are polar bears.
8. Polar bears are white in color.
9. Output: "White"

**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?**

Ans. Algrorithm

1. Define two towns, A and B, separated by 3 km.
2. Town A has 100 students, and Town B has 50 students.
3. The school should be placed in an inverse ratio of student numbers.
4. Since 100:50 simplifies to 2:1, divide the distance accordingly.
5. Distance from Town A = (2/3) × 3 km = 2 km.
6. Distance from Town B = (1/3) × 3 km = 1 km.
7. Output: "The school should be built 1 km from Town A (or 2 km from Town B) for minimal travel distance."

**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**

Ans. Algorithm

1. The traveler has **N** links and must pay 1 link per day.
2. Only one broken link is allowed at a time.
3. To minimize cuts, use a binary system approach.
4. For 6 links, cut at positions 1 and 3:
   1. Day 1: Give 1-link.
   2. Day 2: Exchange 1-link for 2-link.
   3. Day 3: Give 1-link + 2-link (total 3).
   4. Day 4: Exchange previous links for 4-link.
   5. Day 5: Give 1-link + 4-link.
   6. Day 6: Give 2-link + 4-link

.

1. Minimum cuts for 6 links = 2.
2. For 100 links, break into power-of-2 pieces: 1, 2, 4, 8, 16, 32, 37.
3. Total cuts = 6.
4. General case: Minimum cuts required = log₂(N) (rounded up).
5. Output: "For N days, the minimum cuts needed is log₂(N) (rounded up)."

**4. Rearranging "new door" into One Word**

Ans. Algorithm

1. Input the phrase "new door".
2. Remove spaces → "newdoor".
3. Convert to lowercase.
4. Generate all possible anagrams of "newdoor".
5. Check if "oneword" is a valid rearrangement.
6. If yes, print: "Rearranged successfully."
7. Otherwise, print: "No valid rearrangement found."
8. Output: "oneword".

**5. Divide and Conquer Sorting (Merge Sort)**

Ans. Algorithm

1. If the array has more than one element:
   * Divide it into two halves.
   * Left half: First n/2 elements.
   * Right half: Last n/2 elements.
2. Recursively sort each half.
3. Merge sorted halves.
4. Example with [6, 5, 1, 4, 3, 2]:
   * Divide: [6, 5, 1] and [4, 3, 2].
   * Divide further: [6, 5], [1], [4, 3], [2].
   * Break down further: [6], [5], [4], [3].
5. Merge 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, 2, 3, 4, 5, 6].
6. Output: [1, 2, 3, 4, 5, 6].

**6.Draw flowchart for calculating simple interest**

Ans.

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

