

5) The Pigeonhole principle states that if a function f has n distinct inputs but less than n distinct outputs then there are two inputs a and b such that $a \neq b$ and $f(a) = f(b)$. Write a C program to find the values a and b for which the range values are equal.

• Fully implementation focused problem. Done on P.C.

```
1. #include <stdio.h>
2. #define MAXSIZE 100
3. int f(int x) {
4.     return x/5; // example function with 5 possible outputs
5. }
6. int main()
7. {
8.     int arr[MAXSIZE], fval[MAXSIZE];
9.     int n, i;
10.    printf("Enter no. of distinct inputs (n): ");
11.    scanf("%d", &n);
12.    if (n > MAXSIZE) {
13.        printf("Too many inputs (%d)", n);
14.        return 1;
15.    }
16.    printf("Enter %d distinct integer inputs: ", n);
17.    for (i = 0; i < n; i++) {
18.        scanf("%d", &arr[i]);
19.        fval[i] = f(arr[i]);
20.    }
21.    for (i = 0; i < n; i++) {
22.        for (j = i + 1; j < n; j++) {
23.            if (fval[i] == fval[j]) {
24.                printf("Found: f(%d) = f(%d) = %d", arr[i], arr[j], fval[i]);
25.                return 0;
26.            }
27.        }
28.    }
29.    printf("No duplicates found in range values. Pigeonhole Principle not violated (this should not happen if n > no. of possible outputs).");
30.    return 0;
31. }
```