⇒ Space Complexity = (depends on size of input / no. of variables used)

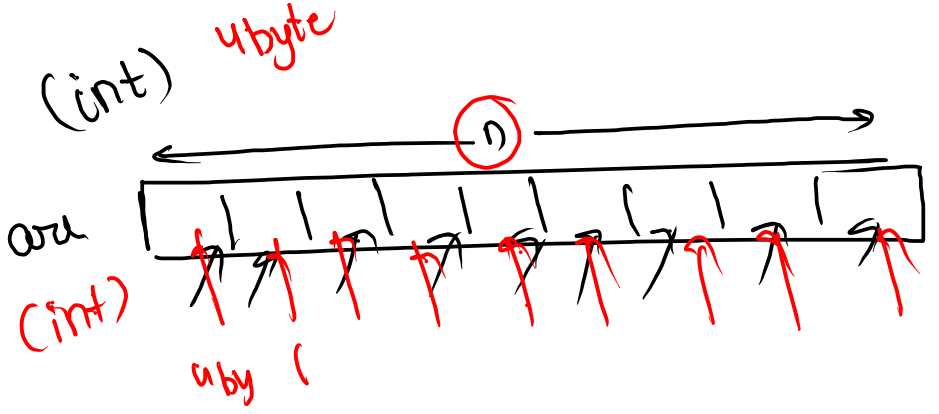
```
public static void main() {
    Scanner scn = ____________
1   int n = scn. - - - -
n   int[] arr = new int[n];
    for(int i=0; i<n; i++){
        Syso("Hi");
    }
}
```

no. of variables :-

n + 1 + 1

S.C = O(n+2)

≅ O(n)



Q2

You are screen sharing | Stop Share

```java
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int arr1[] = new int[n];
        for (int i = 0; i < n; i++) {
            arr1[i] = scn.nextInt();
        }

        int m = scn.nextInt();
        int arr2[] = new int[m];
        for (int i = 0; i < m; i++) {
            arr2[i] = scn.nextInt();
        }

        // main logic
        isIdentical(arr1, arr2, n, m);
    }

    public static void isIdentical(int[] arr1, int[] arr2, int n, int m) {
        if ( n == m ) {

            for (int i = 0; i < n; i++) {
                if ( arr1[i] != arr2[i] ) {
                    System.out.println(false);
                    return;
                }
            }
            System.out.println(true);

        } else {
            System.out.println(false);
        }
    }
}
```

no. of spaces = $7 + n + m + n + m$

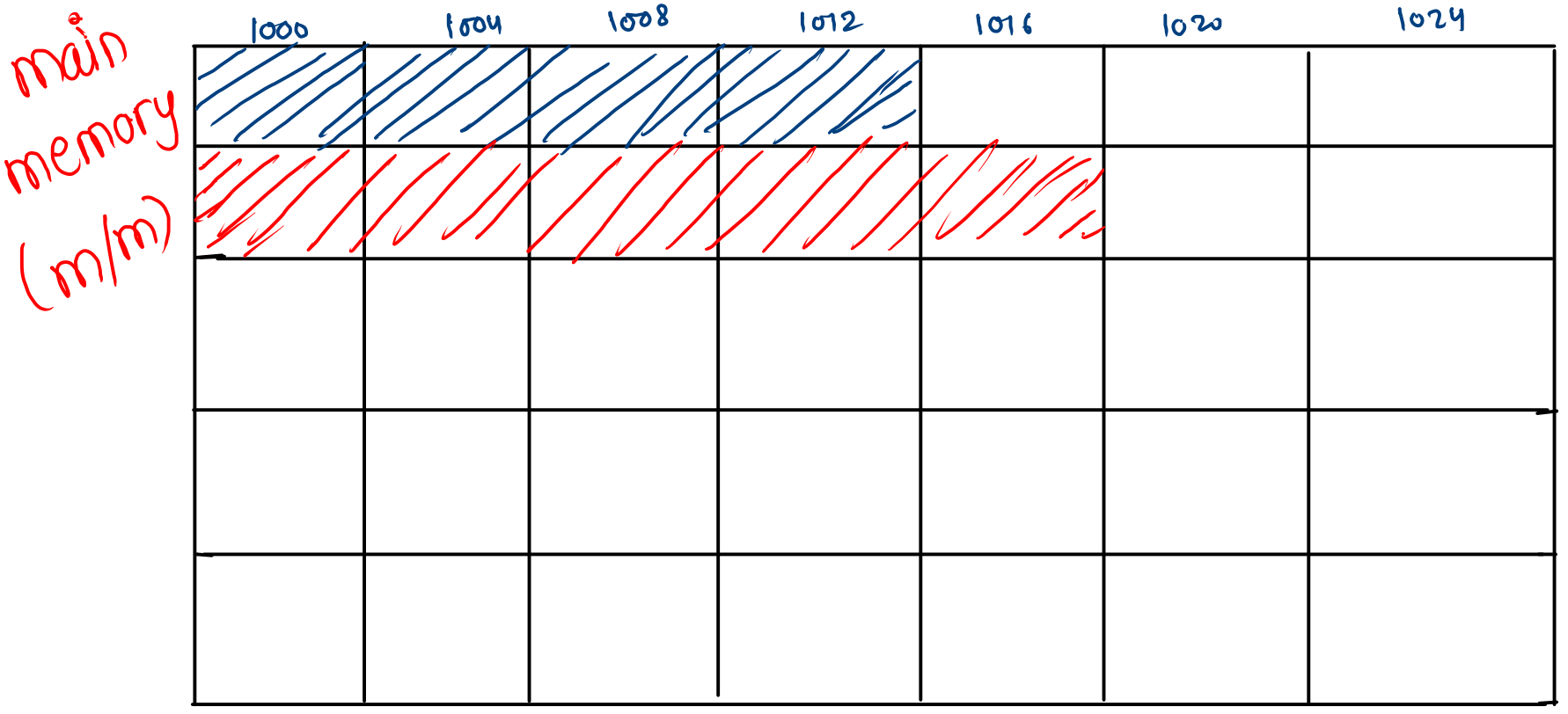
$= 2(n+m) + 7$

S.C = $O(2*(n+m)+7)$

S.C $\cong O(n+m)$

operations :- N

T.C = $O(N)$



n = 4

int [] arr = new int[n];

n = 5

int arr1

int → 4byte → ∞ → 1 Tb

500 Gb

{ disadvantage of array

```
main() {
    1 → int n = ——————
    n → int[] arr = new int[n];
}
```
} n spaces

```
    1 → int m = ——————
    m → int[] arr1 = new int[m];
}
```
} m space

spaces :- $n + 1 + m + 1$

$\Rightarrow$ $n + m + 2$

S.C. $\Rightarrow O(n+m)$

ex 1

```java
public static void main(String[] args) {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be r
    Scanner scn = new Scanner(System.in);
    int x = scn.nextInt();
    int y = scn.nextInt();

    // main logic
    solve(x, y);    // function calling
}

public static void solve(int x, int y) {    // function declaration
    int c = x;
    System.out.println("c = " + c);

    x = y;
    System.out.println("x = " + x);

    y = c;
    System.out.println("y = " + y);

    System.out.println("x = " + x);
    System.out.println("y = " + y);
}
```

no. of operations = 5

T.C = O(1)

no. of spaces = 5

S.C = O(1)

```java
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int[] arr = new int[n];
        for (int i = 0; i < n; i++) {
            arr[i] = scn.nextInt();
        }

        for (int i = 0; i < n; i++) {
            if ( arr[i] > 0 ) {
                arr[i] = 1;
            } else if ( arr[i] < 0 ) {
                arr[i] = -1;
            } else if ( arr[i] == 0 ) {
                arr[i] = 0;
            }
        }

        for (int i = 0; i < n; i++) {
            System.out.print(arr[i] + " ");
        }
    }
}
```

T.C = O(n)

S.C = O(n)

1 approch

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int[] arr = new int[n];
        for (int i = 0; i < n; i++) {
            arr[i] = scn.nextInt();
        }

        for (int i = 0; i < n; i++) {
            int ans = 1;
            for (int j = 0; j < n; j++) {
                if (i != j) {
                    ans = ans * arr[j];
                }
            }
            System.out.println(ans);
        }
    }
}
```

S.C = O(n)

T.C = $O(n^2)$

2 approch

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }

    solve(arr, n);
}

public static void solve(int[] arr, int n) {
    int prod = 1;
    for (int i = 0; i < n; i++) {
        prod = prod * arr[i];
    }

    for (int i = 0; i < n; i++) {
        int ans = prod / arr[i];
        System.out.println(ans);
    }
}
```

S.C = O(n)

T.C = O(n)