

৫২ টি প্রোগ্রামিং সমস্যা ও সমাধান

এক নজরে ৫২ টি প্রোগ্রামিং সমস্যার নাম

- | | |
|--|----------------------------------|
| সমস্যা ১ – জোড়-বিজোর ১ | সমস্যা ২৭ – আমস্ট্রিং সংখ্যা |
| সমস্যা ২ – জোড়-বিজোর ২ | সমস্যা ২৮ – এলোমেলো সংখ্যা |
| সমস্যা ৩ – অধোগামী সংখ্যা | সমস্যা ২৯ – চিহ্ন পরিচয় |
| সমস্যা ৪ – ভাজক | সমস্যা ৩০ – যোগ্য সংখ্যা-১ |
| সমস্যা ৫ – বাস্তু ১ | সমস্যা ৩১ – যোগ্য সংখ্যা-২ |
| সমস্যা ৬ – যোগফল নির্ণয় | সমস্যা ৩২ – x এর গুণিতক |
| সমস্যা ৭ – সংখ্যা গণনা | সমস্যা ৩৩ – বিভাজনসাধ্য ১ |
| সমস্যা ৮ – ছোট থেকে বড় | সমস্যা ৩৪ – বিভাজনসাধ্য ২ |
| সমস্যা ৯ – পূর্ণবর্গ সংখ্যা | সমস্যা ৩৫ – বৃত্তের বাইরে |
| সমস্যা ১০ – রান রেট-১ | সমস্যা ৩৬ – শব্দ সাজানো |
| সমস্যা ১১ – গৌণিক বা ফ্যাক্টোরিয়াল | সমস্যা ৩৭ – সংখ্যা বিপর্যয় |
| সমস্যা ১২ – ফ্যাক্টোরিয়াল ১০০ | সমস্যা ৩৮ – হীরক রাজ্য |
| সমস্যা ১৩ – টমি মিয়ার প্রোবাবিলিটি | সমস্যা ৩৯ – প্যালিনড্রোম |
| সমস্যা ১৪ – অক্ষরের ঘনঘটা | সমস্যা ৪০ – ধারার যোগফল-১ |
| সমস্যা ১৫ – অক্ষর গণনা | সমস্যা ৪১ – ধারার যোগফল-২ |
| সমস্যা ১৬ – শব্দ বিপর্যয় | সমস্যা ৪২ – ধারার যোগফল-৩ |
| সমস্যা ১৭ – স্বরবর্ণ গণনা | সমস্যা ৪৩ – হিসাবকিতাব |
| সমস্যা ১৮ – স্বরবর্ণ – ব্যঞ্জনবর্ণ | সমস্যা ৪৪ – প্যাসকেলের ত্রিভুজ-১ |
| সমস্যা ১৯ – শব্দ গণনা-১ | সমস্যা ৪৫ – প্যাসকেলের ত্রিভুজ-২ |
| সমস্যা ২০ – শব্দ গণনা-২ | সমস্যা ৪৬ – ত্রিভুজের ক্ষেত্রফল |
| সমস্যা ২১ – উলটে দেখা | সমস্যা ৪৭ – অ্যারের জোট |
| সমস্যা ২২ – মৌলিক সংখ্যা | সমস্যা ৪৮ – নিখোঁজ সংখ্যা |
| সমস্যা ২৩ – বর্ণমালা থেকে সংখ্যা | সমস্যা ৪৯ – মৌলিক কি না |
| সমস্যা ২৪ – একান্তর উপাদান | সমস্যা ৫০ – লেফট-রাইট |
| সমস্যা ২৫ – লগিষ্ট সাধারণ গুণীয়ক(ল.সা.গু) | সমস্যা ৫১ – খোঁজ দ্যা সার্চ-১ |
| সমস্যা ২৬ – এলিয়েন গুপি | সমস্যা ৫২ – খোঁজ দ্যা সার্চ-২ |

সমস্যা ১ – জোড়-বিজোর ১

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int num;
        scanf("%d", &num);
        if (num % 2 == 0) printf("even\n");
        else printf("odd\n");
    }
}
```

Output:
3
100
even
0
even
1111
odd

এখানে num এর মান হতে পারবে 0-2147483647 পর্যন্ত।
তারমানে int টাইপের ভেরিয়েবল ব্যবহার করলেই হবে,
কারণ int টাইপের ভেরিয়েবল এর সর্বোচ্চ মান 2147483647.

সমস্যা ২ – জোড়-বিজোর ২

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        char num[101];
        scanf("%s", num);
        int length = strlen(num);
        char lastcharacter = num[length - 1];
        int lastnumber = lastcharacter - '0';

        if (lastnumber % 2 == 0) printf("even\n");
        else printf("odd\n");
    }
}
```

Output:
3
1000
even
0
even
1111
Odd

- সিং এর ক্ষেত্রে & ব্যবহার না করাই উত্তম। `scanf("%s", num);` তবে ব্যবহার করলেও সমস্যা নাই।
- এখানে সিং ব্যবহার করা হয়েছে কারন একটি সংখ্যায় সর্বোচ্চ 100 টি digit থাকতে পারে। যা অনেক অনেক বড় সংখ্যা।
- `int/sign int/unsigned int- variable` হচ্ছে 32 বিট বা 2^{32} বা 4294967296.
- `long long int- variable` হচ্ছে 64 বিট। এতে 2^{64} বা 19 টি digit রাখা যাবে।

সমস্যা ৩ – অধোগামী সংখ্যা

```
#include <stdio.h>

int main()
{
    int count = 0;
    for (int i = 1000; i >= 1; i--)
    {
        printf("%d\t", i);
        count = count + 1;

        if (count == 5) {
            printf("\n");
            count = 0;
        }
    }
}
```

Output:				
1000	999	998	997	996
995	994	993	992	991
990	989	988	987	986
-	-	-	-	-
10	9	8	7	6
5	4	3	2	1

if (count == 5) বা, if (count % 5 == 0) এভাবে করলেও হতো।

সমস্যা ৪ - ভাজক

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int x = 1; x <= t; x++)
    {
        int n;
        scanf("%d", &n);
        printf("Case %d:", x);
        for (int i = 1; i <= n; i++)
        {
            if (n % i == 0)
                printf(" %d", i);
        }
        printf("\n");
    }
}
```

Output:
3
6
Case 1 : 1 2 3 6
15
Case 2 : 1 3 5 15
23
Case 3 : 1 23

সমস্যা ৫ – বাক্স ১

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int x = 1; x <= t; x++)
    {
        int n;
        scanf("%d", &n);
        for (int i = 1; i <= n; i++)
        {
            for (int j = 1; j <= n; j++)
            {
                printf("*");
            }
            printf("\n");
        }
        printf("\n");
    }
}
```

Output:

```
3
1
*

3
***
***
***

5
*****
*****
*****
*****
*****
```

Second printf("\n");
এটাকে এভাবে লিখলেও হবে,
if (x != t)
printf("\n");

সমস্যা ৬ – যোগফল নির্ণয়

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int x = 1; x <= t; x++)
    {
        int n;
        scanf("%d", &n);
        int temp = n;
        int sum = 0, rem;
        while (temp != 0)
        {
            rem = temp % 10;
            temp = temp / 10000;
            sum = sum + rem;
        }
        printf("%d\n", sum);
    }
}
```

বিকল্প পদ্ধতিঃ

```
int n;
scanf("%d", &n);
int sum = 0;
sum = sum + n % 10;
sum = sum + n / 10000;
printf("%d", sum);
```

Output:

```
3
12345
6
56789
14
14310
1
```

যেহেতু ৫ টা সংখ্যা নিয়ে প্রোগ্রাম টা করা হয়েছে তাই 10000 দিয়েছি।
৩ টা সংখ্যা হলে 100 দিতাম। ৪ টা হলে 1000 দিতাম।

সমস্যা ৭ – সংখ্যা গণনা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int x = 1; x <= t; x++)
    {
        char str[100001];
        gets(str);
        /*scanf("%[^\\n]", str);*/
        int i = 0;
        int count = 0, flag = 0;

        /*This loop will run, untill it reach at the end of the string*/
        for (int i = 0; str[i] != '\\0'; i++)
        {
            if (((str[i] >= '0' && str[i] <= '9') || str[i] == '-') && flag == 0) {
                count = count + 1;
                flag = 1;
            }
            else if (!(str[i] >= '0' && str[i] <= '9') || str[i] == '-') && flag == 1)
                flag = 0;
        }
        printf("%d\\n", count);
    }
}
```

<p>Output:</p> <pre>3 1 - 2 10000 - 50 20 7 455 7 9 1 - 98 876 65 3</pre>	<p>#include <stdio.h></p> <pre>int main() { char str[151]; gets(str); int word = 1; for (int i = 0; str[i] != '\\0'; i++) { if (str[i] == ' ' && str[i + 1] != ' ') word++; } printf("%d", word); }</pre> <p>Output:</p> <pre>Golam Kibria Ezaz 3</pre>	শব্দ গণনা
---	--	-----------

সমস্যা ৮ – ছোট থেকে বড়

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int n1, n2, n3;
        scanf("%d %d %d", &n1, &n2, &n3);
        int larger, middle, smaller;
        if (n1 > n2 && n1 > n3)        larger = n1;
        else if (n2 > n1 && n2 > n3) larger = n2;
        else larger = n3;
        if (n1 < n2 && n1 < n3) smaller = n1;
        else if (n2 < n1 && n2 < n3) smaller = n2;
        else smaller = n3;

        middle = n1 + n2 + n3 - larger - smaller;

        printf("Case %d: %d %d %d\n", i, smaller, middle, larger);
    }
}
```

```
Output:
3
3 2 1
Case 1 : 1 2 3
1 2 3
Case 2 : 1 2 3
10 5 6
Case 3 : 5 6 10
```

সমস্যা – ছোট থেকে বড় (বিকল্প সমাধান)

/*অ্যারের সাহায্যে*/

```
#include <stdio.h>

int main()
{
    int array[10];
    int n;
    printf("How many number : ");
    scanf("%d", &n);
    printf("Enter %d number : \n", n);
    for (int i = 0; i < n; i++) {
        scanf("%d", &array[i]);
    }
    int temp;
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (array[i] > array[j])
            {
                temp = array[i];
                array[i] = array[j];
                array[j] = temp;
            }
        }
    }
    for (int i = 0; i < n; i++) {
        printf("%d ", array[i]);
    }
}
```

Output:
How many number : 5
Enter 5 number :
34 12 67 99 35
12 34 35 67 99

যদি বড় থেকে ছোট ক্রমানুসারে সাজাতে বলতো তখন শুধু লাস্টের লুপটা এইভাবে পরিবর্তন করলেই হবে

```
for (int i = n - 1; i >= 0; i--) {
    printf("%d ", array[i]);
}
```

সমস্যা ৯ – পূর্ণবর্গ সংখ্যা

```
#include <stdio.h>
#include <math.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int n, a;
        scanf("%d", &n);
        a = sqrt(n);
        if (a * a == n)
            printf("Yes\n");
        else printf("No\n");
    }
}
```

Output:

```
3
16
Yes
18
No
196
Yes
```

- পূর্ণবর্গ সংখ্যা হচ্ছে সেই সব সংখ্যা যাদের বর্গমূল একটি পূর্ণসংখ্যা।
- অবশ্যই অবশ্যই math.h লিখতে হবে, না হলে wrong answer দেখাবে।
- ❖ ceil(2.1) return value 3.0
- ❖ ceil(2.8) return value 3.0
- ❖ floor(2.1) return value 2
- ❖ floor(2.8) return value 2

```
#include <stdio.h>
#include <math.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        double n;
        scanf("%lf", &n);
        double a = sqrt(n);
        if (ceil(a) == floor(a))
            printf("Yes\n");
        else printf("No\n");
    }
}
```

সমস্যা ১০ – রান রেট-১

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int orakorcy, amrakorci, ballbaki;
        scanf("%d %d %d", &orakorcy, &amrakorci, &ballbaki);
        int ballkhelci = 300 - ballbaki;
        double crr, rrr;

        crr = ((double)amrakorci / (double)ballkhelci) * 6;
        rrr = ((double)(orakorcy - amrakorci + 1) / (double)ballbaki) * 6;

        if (amrakorci > orakorcy)
            rrr = 0.0;

        printf("%.21f %.21f\n", crr, rrr);
    }
}
```

Output:
2
300 294 6
6.00 7.00
200 100 100
3.00 6.06

crr এবং rrr বের করার সূত্র দুইটা অবশ্যই মনে রাখতে হবে।

সমস্যা ১১ – গৌণিক বা ফ্যাক্টোরিয়াল

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int x = 1; x <= t; x++)
    {
        int n;
        scanf("%d", &n);
        long long int factorial = 1;
        for (int i = 2; i <= n; i++)
        {
            factorial = factorial * i;
        }
        printf("%lld\n", factorial);
    }
}
```

Output:
3
6
720
10
3628800
15
1307674368000

15 এর factorial অনেক বড় সংখ্যা তাই long long int ডাটা টাইপ ব্যবহার করা হয়েছে।
long long int- variable হচ্ছে 64 বিট। এতে 2^{64} বা 19 টি digit রাখা যাবে।

সমস্যা ১২ – ফ্যাক্টোরিয়াল ১০০

/*একটি সংখ্যার ফ্যাক্টোরিয়াল এর শেষে কতটি ০ আছে তা বের করতে হবে*/

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n;
        scanf("%d", &n);
        int count = 0;
        int x = 5, a;
        while (n >= x)
        {
            a = n / x;
            x = x * 5;
            count = count + 1;
        }
        printf("%d\n", count);
    }
}
```

Output:

3
6
1
15
1
100
2

এই প্রোগ্রামটিতে ৫ দিয়ে কম্পেয়ার করা হয়েছে।

কারণ $৫০০ = ৫ \times ১০ \times ১০$

তারমানে দুইটি ১০ আছে তাই ০ দুটি।

```
#include <stdio.h>
```

বিকল্প সমাধান

```
int main()
{
    int n;
    scanf("%d", &n);
    int count = 0, rem;
    while (n > 0) {
        rem = n % 10;
        n = n / 10;
        if (rem == 0) count++;
        else break;
    }
    printf("%d\n", count);
}
```

সমস্যা ১৩ – টমি মিলার প্রোবাবিলিটি

সমস্যা ১৪ – অক্ষরের ঘনঘটা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        char firstline[10001];
        char secondline[2];

        /*যেহেতু স্পেস থাকবে তাই এইভাবে স্কেন করাই উত্তম
        অবশ্যই আগে একটা স্পেস দিতে হবে।*/
        scanf(" %[^\n]", firstline);
        scanf("%s", secondline);

        int count = 0;
        int length = strlen(firstline);
        for (int i = 0; i < length; i++)
        {
            if (secondline[0] == firstline[i])
                count++;
        }
        if (count > 0)
            printf("Occurance of %c in %s = %d\n", secondline[0], firstline, count);
        else
            printf("%c is not found\n", secondline[0]);
    }
}
```

Output:
2
hello world
1
Occurance of l in hello world = 3
golam kibria
a
Occurance of a in golam kibria = 2

scanf(" %[^\n]", s);
% এর আগে স্পেস দেয়ার কারন হলো আগের লাইনের শেষে যে নিউলাইন ক্যারেক্টার(\n) আছে সেটি যেন এর ভেতর ঠুকে যায়।

সমস্যা ১৫ – অক্ষর গণনা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        char s[1001];
        int count[26];
        scanf(" %[^\n]", s);
        int length = strlen(s);
        memset(count, 0, sizeof count);
        /*Syntax of memset (N, o, sizeof N)*/
        for (int i = 0; i < length; i++)
        {
            if (s[i] >= 'a' && s[i] <= 'z')
                count[s[i] - 'a']++;
        }
        for (int i = 0; i < 26; i++)
        {
            if (count[i] != 0)
                printf("%c = %d\n", 'a' + i, count[i]);
        }
        printf("\n");
    }
}
```

Output:

```
2
hello
e = 1
h = 1
l = 2
o = 1
```

```
baby
a = 1
b = 2
y = 1
```

When $i = 0$

'a'+i = 97+0 = 97

So, %d = 97 but %c = 'a'

Function `memset()` is a library function of `string.h`. It is used to fill a block of memory with given / particular value. It is used when you want to fill all or some of the blocks of the memory with a particular value.

সমস্যা ১৬ – শব্দ বিপর্যয়

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int h = 0; h <= t; h++)
    {
        char str[1001];
        gets(str);
        /*scanf(" %[^\n]", str);*/
        int length = strlen(str);
        int temp = 0;

        for (int i = 0; i <= length; i++)
        {
            if (str[i] == ' ' || str[i] == '\0')
            {
                if (temp != 0)
                {
                    printf(" ");
                }
                int last = i - 1;
                while (last >= temp)
                {
                    printf("%c", str[last]);
                    last--;
                }
                temp = i;
            }
        }
        printf("\n");
    }
}
```

Output:
2
This is a test
sihT si a tset
Hello world
olleH dlrow

সমস্যা ১৭ – স্বরবর্ণ গণনা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int h = 1; h <= t; h++)
    {
        char str[1001];
        gets(str);
        int count = 0;
        int length = strlen(str);
        for (int i = 0; i < length; i++) {
            if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' ||
str[i] == 'o' || str[i] == 'u' || str[i] == 'A' || str[i] == 'E' ||
str[i] == 'I' || str[i] == 'O' || str[i] == 'U') {
                count++;
            }
        }
        printf("Number of vowels : %d\n", count);
    }
}
```

Output:
2
i am a programmer
Number of vowels : 6
hello world
Number of vowels : 3

for (int i = 0; i < length; i++)
এই লাইন টা এইভাবে লিখলেও হতো,
for (int i = 0; str[i]!='\0'; i++)

সমস্যা ১৮ – স্বরবর্ণ – ব্যঞ্জনবর্ণ

```
#include <stdio.h>
#include <string.h>

int main()
{
    int n;
    scanf("%d", &n);
    getchar();
    for (int h = 1; h <= n; h++)
    {
        char str[1001];
        gets(str);
        char vowel[1001];
        char consonent[1001];
        int v = 0, c = 0;

        for (int i = 0; str[i] != '\0'; i++)
        {
            if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z'))
            {
                if (str[i] == 'a' || str[i] == 'A' || str[i] == 'e' || str[i] == 'E'
|| str[i] == 'i' || str[i] == 'I' || str[i] == 'o' || str[i] == 'O' || str[i] == 'u'
|| str[i] == 'U') {
                    vowel[v] = str[i];
                    v++;
                }
                else {
                    consonent[c] = str[i];
                    c++;
                }
            }
        }
        /*Terminating string by NULL charecter*/
        vowel[v] = '\0';
        consonent[c] = '\0';

        printf("%s\n", vowel);
        printf("%s\n", consonent);
    }
}
```

Output:

2

this is very easy

ieea

thssvrys

it is a rainy sunday

iaaiua

tsrnysndy

```
for (int i = 0; str[i] != '\0'; i++)
{ এর নিচে এই লাইন টা দিলেও হতো
    if (str[i] == ' ') {
        i++;
        continue;
    }
```

সমস্যা ১৯ – শব্দ গণনা-১

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int i = 1; i <= t; i++)
    {
        char str[10001];
        gets(str);
        int length = strlen(str);
        int word = 1;
        for (int i = 0; i < length; i++)
        {
            if (str[i] == ' ' && str[i + 1] != ' ')
                word++;
        }
        printf("Total word : %d\n", word);
    }
}
```

Output:
2
Hello world
Total word : 2
Dhaka is the capital of bangladesh
Total word : 6

সমস্যা ২০ - শব্দ গণনা-২

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int i = 1; i <= t; i++)
    {
        char str[10001];
        gets(str);
        int length = strlen(str);
        int word = 1;
        for (int i = 0; i < length; i++)
        {
            if (str[i] == ' ' && str[i + 1] != ' ')
                word++;
        }
        printf("Total word : %d\n", word);
    }
}
```

Output:
2
Hurrah!was heard on all sides
Total word : 6
Hello, I'm a broker
Total word : 4

সমস্যা ২১ – উলটে দেখা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int h = 1; h <= t; h++)
    {
        char str[1001];
        gets(str);
        int length = strlen(str);
        for (int i = length - 1; i >= 0; i--)
        {
            printf("%c", str[i]);
        }
        printf("\n");
    }
}
```

Output:
3
string
gnirts
programming
gnimmargorp
aruna
anura

সমস্যা ২২ – মৌলিক সংখ্যা

```
#include<stdio.h>
#include<math.h>

/*this function will check wheather it's prime or not.
if the number is prime then this function will simply return 1
else it will return 0 and we will add the return value to count*/
int primeornot(int x)          /*x = a X b ; a ও b এর যেকোনো একটি সংখ্যা অবশ্যই
                                x এর বর্গমূলের সমান বা তার চেয়ে ছোট হবে*/
{
    int i;
    for (i = 2; i <= sqrt(x); i++)
    {
        if (x % i == 0) return 0;
    }
    return 1;
}

int main()
{
    int t, i;
    scanf("%d", &t);
    for (i = 1; i<=t; i++)
    {
        int a, b;
        scanf("%d %d", &a, &b);
        int count = 0;
        //creating a loop from a to b
        while (a <= b)
        {
            /*1 and 0 is not prime number so
            if a=1 or a=0 we won't check prime or not*/
            if (a == 1 || a == 0)
            {
                a++;
                continue;
            }
            //adding returning value to countt
            count = count + primeornot(a);
            a++;
        }
        printf("%d\n", count);
    }
    return 0;
}
```

সমস্যা ২২ – মৌলিক সংখ্যা

/*আগের টার তুলনায় এই পদ্ধতিটা সহ্য*/

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int low, high, i, flag;
        printf("Enter two numbers : ");
        scanf("%d %d", &low, &high);
        printf("prime numbers : ");
        int count = 0;

        while (low <= high)
        {
            flag = 0;

            if (low == 1 || low == 0) {
                low++;
                continue;
            }

            /*If low is a non-prime number, flag will be 1*/
            for (i = 2; i <= low / 2; i++)
            {
                if (low % i == 0) {
                    flag = 1;
                    break;
                }
            }

            if (flag == 0)
            {
                count++;
                printf("%d ", low);
            }
            low++;
        }
        printf("\nSo, total prime number %d\n", count);
    }
}
```

Output:
2
Enter two numbers : 1 5
prime numbers : 2 3 5
So, total prime number 3
Enter two numbers : 25 50
prime numbers : 29 31 37 41 43 47
So, total prime number 6

সমস্যা ২৩ – বর্ণমালা থেকে সংখ্যা

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int h = 1; h <= t; h++)
    {
        char str[101];
        gets(str);
        int length = strlen(str);

        /*int count = 0;*/

        for (int i = 0; i < length; i++) {
            if (str[i] >= 'A' && str[i] <= 'Z') {
                printf("%d", str[i] - 64);
            }

            /*if (str[i] == 'A' || str[i] == 'E' || str[i] == 'I'
            || str[i] == 'O' || str[i] == 'U')
                count++;*/

        }
        /*printf("\nNumber of vowels : %d\n", count);*/
        printf("\n");
    }
}
```

Output:
3
ABZ
1226
ZYB
26252
CODING
31549147

যদি বলতো কতগুলো vowel আছে তা নির্ণয় করো
তখন কমেন্ট এর গুলো দিয়ে দিলেই হবে।

সমস্যা ২৪ – একান্তর উপাদান

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int num[101];
        int n;
        scanf("%d", &n);
        for (int i = 0; i < n; i++)
        {
            scanf("%d", &num[i]);
        }
        for (int i = 0; i < n; i = i + 2)
        {
            printf("%d ", num[i]);
        }
        printf("\n");
    }
}
```

Output:

```
2
5 1 2 3 4 5
1 3 5
10 1 4 55 66 22 0 76 11 23 78
1 55 22 76 23
```

সমস্যা ২৫ – লগিষ্ট সাধারণ গুণীয়ক(ল.সা.গু)

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int num1, num2, n1, n2, rem;
        int lcm, gcd;

        /*printf("Please enter two number : ");*/
        scanf("%d %d", &num1, &num2);

        n1 = num1;
        n2 = num2;

        while (n2 != 0)
        {
            rem = n1 % n2;
            n1 = n2;
            n2 = rem;
        }
        gcd = n1;
        lcm = (num1 * num2) / gcd;
        /*printf("GCD = %d\n", gcd);\n*/
        printf("LCM = %d\n", lcm);
    }
}
```

Output:
2
30 15
LCM = 30
12 16
LCM = 48

n1 % n2 = rem
30 % 15 = 0
15 % 0 (n2 is 0, so the loop can not execute)
So, gcd = n1 = 15;
And, lcm = (30 * 15) / 15 = 30;

সমস্যা ২৬ – এলিয়েন গুপি

/*একদিনে তার সরবরাহকৃত খাবারের অর্ধেক খেয়ে ফেলে।

অর্থাৎ, যদি x পরিমাণ খাবার নিয়ে আসে তাহলে ১ দিনে খায় $x/2$.

প্রশ্ন হচ্ছে তাহলে তার খাবার শেষ করতে কতদিন লাগবে সেটা বের করা।*/

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        double x;
        scanf("%lf", &x);
        int count = 0;
        while (x > 1.0)
        {
            x = x / 2;
            count++;
        }
        printf("%d days\n", count);
    }
}
```

Output:

3

40

6 days

200

8 days

300

9 days

সমস্যা ২৭ – আর্মস্ট্রং সংখ্যা

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int n;
        scanf("%d", &n);
        int rem, sum = 0;
        int temp = n;
        while (temp != 0)
        {
            rem = temp % 10;
            temp = temp / 10;
            sum = sum + rem * rem * rem;
        }
        if (sum == n)
            printf("%d is an armstrong numebr\n", n);
        else
            printf("%d is not an armstrong number\n", n);
    }
}
```

Output:

```
3
100
100 is not an armstrong number
153
153 is an armstrong numebr
371
371 is an armstrong numebr
```

Example of Armstrong number:

$$153 = 1^3 + 5^3 + 3^3$$

এখানে 3 নিয়েছি কারন 153 এর মধ্যে 3 টি অঙ্ক রয়েছে।

সমস্যা ২৮ – এলোমেলো সংখ্যা

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n;
        scanf("%d", &n);
        int max = -100;
        int flag = 0;
        for (int i = 1; i <= n; i++)
        {
            int num;
            scanf("%d", &num);
            if (num >= max) max = num;
            else {
                flag = 1;
                break;
            }
        }
        if (flag == 0) printf("YES\n");
        else printf("NO\n");
    }
}
```

Output:
2
5
1 2 3 4 5
YES
10
1 2 3 4 5 6 7 9 10 8
NO

সমস্যা ২৯ – চিহ্ন পরিচয়

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        char ch;
        getchar();
        ch = getchar();

        if (ch >= 'a' && ch <= 'z')
            printf("Lowercase Character");
        else if (ch >= 'A' && ch <= 'Z')
            printf("Uppercase Character");
        else if (ch >= '0' && ch <= '9')
            printf("Numerical Digit");
        else
            printf("Special Character");
        printf("\n");
    }
}
```

```
Output;
4
a
Lowercase Character
A
Uppercase Character
5
Numerical Digit
;
Special Character
```

সমস্যা ৩০ – যোগ্য সংখ্যা-১

- Perfect number/যোগ্য সংখ্যা?
- ✓ কোনো একটি সংখ্যা এবং ঐ সংখ্যাটির ভাজকগুলোর যোগফল যদি পরস্পর সমান হয় তাহলেই তাকে perfect number বলা যাবে।

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int num;
        scanf("%d", &num);
        int sum = 0;
        for (int i = 1; i <= num / 2; i++)
        {
            if (num % i == 0)
                sum = sum + i;
        }
        if (sum == num)
            printf("YES, %d is perfect number\n", num);
        else
            printf("NO, %d is not a perfect number\n", num);
    }
}
```

Output:

```
3
6
YES, 6 is perfect number
28
YES, 28 is perfect number
30
NO, 30 is not a perfect number
```

যদি $(num \leq 2^{64}-1)$ বলে তখন unsigned long long ব্যবহার করতে হবে।
এর format specifier হলো %llu
An unsigned long long occupies 8 bytes of memory;
It stores an integer from 0 to $2^{64} - 1$,

সমস্যা ৩১ – যোগ্য সংখ্যা-২

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int num;
        scanf("%d", &num);
        if (num >= 6)printf("6\n");
        if (num >= 28)printf("28\n");
        if (num >= 496)printf("496\n");
        if (num >= 8128)printf("8128\n");
        if (num >= 33550336)printf("33550336\n");
        printf("\n");
    }
}
```

Output:

```
2
100
6
28

500
6
28
496
```

- int num এর পরিবর্তে long long int num ব্যবহার করা যেতে পারে ।
তখন, format specifier হবে %lld
- 1-40000000 এর মধ্যে মাত্র পাঁচটি যোগ্য সংখ্যা আছে
সেগুলো হলো 6, 28, 496, 8128, 33550336

সমস্যা ৩২ – x এর গুনিতক

/*x থেকে n পর্যন্ত x এর গুনিতক গুলো প্রিন্ট করতে হবে। যদি x>n হয় তাহলে invalid প্রিন্ট করতে হবে।*/

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int x, n;
        scanf("%d %d", &x, &n);
        for (int i = 1; i <= n; i++)
        {
            if (i % x == 0)
                printf("%d ", i);
        }
        if (x > n) printf("Invalid\n");
        printf("\n");
    }
}
```

Output:

```
3
2 10
2 4 6 8 10
99 1000
99 198 297 396 495 594 693 792 891 990
10 2
Invalid
```

```
for (int i = 1; i <= n; i++)
এই লুপ টা এইভাবেও লিখতে পারতাম
for (int i = x; i <= n; i = i + x)
```

সমস্যা ৩৩ – বিভাজনসাহ্য ১

/*প্রোগ্রামটির আউটপুটে a থেকে b পর্যন্ত যতগুলো সংখ্যা c দ্বারা নিঃশেষে বিভাজ্য সেই সংখ্যা গুলো প্রিন্ট করতে হবে।*/

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int a, b, c;
        scanf("%d %d %d", &a, &b, &c);

        for (int i = a; i <= b; i++)
        {
            if (i % c == 0)
                printf("%d ", i);
        }
        printf("\n");
    }
}
```

Output:
3
2 20 3
3 6 9 12 15 18
50 60 5
50 55 60
55 100 6
60 66 72 78 84 90 96

for (int i = a; i <= b; i++)
এই লুপটাকে এইভাবেও লিখা যেতো
for (int i = 1; i <= b; i = i + 1)

- Int দিয়ে করলেও কোড হয়ে যাবে কিন্তু যদি রেঞ্জ বেশি হয় সেক্ষেত্রে long দিয়ে করাটাই উত্তম।
- long এর format specifier হলো %ld

সমস্যা ৩৪ – বিভাজনসাধ্য ২

/*প্রোগ্রামটির আউটপুটে 1 থেকে c পর্যন্ত যতগুলো সংখ্যা a ও c দ্বারানিঃশেষে বিভাজ্য সেই সংখ্যা গুলো প্রিন্ট করতে হবে।*/

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int a, b, c;
        scanf("%d %d %d", &a, &b, &c);

        for (int i = 1; i <= c; i++)
        {
            if (i % a == 0 && i % b == 0)
                printf("%d ", i);
        }
        printf("\n");
    }
}
```

Output:

```
3
2 3 50
6 12 18 24 30 36 42 48
3 5 50
15 30 45
5 6 100
30 60 90
```

সমস্যা ৩৫ – বৃত্তের বাইরে

```
#include <stdio.h>
#include <math.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        double x1, y1;
        double r;
        double x2, y2;

        scanf("%lf %lf", &x1, &y1);
        scanf("%lf", &r);
        scanf("%lf %lf", &x2, &y2);

        double distance = sqrt((x2 - x1) * (x2 - x1) + (y2 - y1) * (y2 - y1));

        if (r >= distance)
            printf("The point is inside the circle\n");
        else
            printf("The point is not inside the circle\n");
        printf("\n");
    }
}
```

```
Output:
2
1 1
4
10 - 14
The point is not inside the circle

1 1
8
5 6
The point is inside the circle
```

/*যেহেতু sqrt ফাংশনটি ব্যবহার করেছি তাই math.h নামক হেডার ফাইলটি ব্যবহার করা হয়েছে*/

সমস্যা ৩৬ – শব্দ সাজানো

```
/*Alphabetical order*/

#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n;
        scanf("%d", &n);
        char name[10][10], cname[10][10], temp[10];
        for (int i = 0; i < n; i++)
        {
            scanf("%s", name[i]);
            strcpy(cname[i], name[i]);
        }
        for (int i = 0; i < n; i++)
        {
            for (int j = i + 1; j < n; j++)
            {
                if (strcmp(name[i], name[j]) > 0)
                {
                    strcpy(temp, name[i]);
                    strcpy(name[i], name[j]);
                    strcpy(name[j], temp);
                }
            }
        }
        printf("\n");
        for (int i = 0; i < n; i++) {
            printf("%s\n", name[i]);
        }
    }
}
```

Output:

```
1
5
xray
apple
cat
bat
house
```

```
apple
bat
cat
house
xray
```


সমস্যা ৩৭ – সংখ্যা বিপর্যয়

/*একটি সংখ্যাকে উল্টে প্রিন্ট করা*/

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int n;
        scanf("%d", &n); //12
        int sum = 0;
        while (n != 0)
        {
            sum = sum * 10; //0, 20
            sum = sum + n % 10; //0+2=2, 20+1=21
            n = n / 10; //1, 0
        }
        printf("%d\n", sum);
    }
}
```

Output:
3
612
216
1431056
6501341
10041992
29914001

সমস্যা ৩৮ – হীরক রাজ্য

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n, a;
        scanf("%d %d", &n, &a);

        for (int row = 1; row <= n; row++)
        {
            for (int col = 1; col <= row; col++)
            {
                printf("%d ", a);
            }
            printf("\n");
        }
        for (int row = n - 1; row >= 1; row--)
        {
            for (int col = 1; col <= row; col++)
            {
                printf("%d ", a);
            }
            printf("\n");
        }
        printf("\n");
    }
}
```

Output:	5 2
2	2
3 1	2 2
1	2 2 2
1 1	2 2 2 2
1 1 1	2 2 2 2 2
1 1	2 2 2 2
1	2 2 2
	2 2
	2

/*পুরোটাই প্যাটার্ন টাইপ-৩ এর মতো সেম*/

সমস্যা ৩৯ - প্যালিনড্রোম

/*কোনো একটি সংখ্যা প্যালিনড্রোম কিনা তা নির্ণয়*/

```
#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);          //111
    int temp, rem, sum = 0;
    temp = n;

    while (temp != 0)
    {
        rem = temp % 10;      //1, 1, 1
        temp = temp / 10;     //11, 1, 0
        sum = sum * 10 + rem; //1, 11, 111
    }
    if (sum == n)
        printf("Yes! It is palindrome\n");
    else
        printf("Sorry! It is not palindrome\n");
}
```

Output: 121 Yes!It is palindrome
123 Sorry!It is not palindrome
আবার শুধু যদি sum print করতাম তখন reverse হয়ে যেত ।

সমস্যা ৩৯ – প্যালিনড্রোম

/*কোনো একটি স্ট্রিং প্যালিনড্রোম কিনা তা নির্ণয়*/

```
#include <stdio.h>
#include <string.h>

int main()
{
    int t;
    scanf("%d", &t);
    getchar();
    for (int h = 1; h <= t; h++)
    {
        char str[1001];
        gets(str);
        int length = strlen(str);

        char cstr[1001];
        cstr[length] = '\0';
        int i = 0;
        /*সুবিধার জন্য আলাদা একটা ভেরিয়েবল এ রেখে দিলাম*/
        int a = length - 1;

        while (a >= 0)
        {
            cstr[a] = str[i];
            a--; i++;
        }
        /*Checking palindrome or not*/
        if (strcmp(str, cstr) == 0)
            printf("Yes. It is palindrome\n");
        else
            printf("No. It is not palindrome\n");
    }
}
```

Output:
3
wow
Yes.It is palindrome
string
No.It is not palindrome
civic
Yes.It is palindrome

সমস্যা ৪০ – ধারার যোগফল-১

$x^0 + x^1 + x^2 + x^3 + x^4 + \dots + x^k$ এবং k এর মান দেয়া থাকবে। অর্থাৎ লুপটা চলবে 0 থেকে k পর্যন্ত।*/

```
#include <stdio.h>
#include <math.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int i = 1; i <= t; i++)
    {
        int x, k;
        scanf("%d %d", &x, &k);
        int sum = 0;
        for (int i = 0; i <= k; i++)
        {
            sum = sum + pow(x, i);
        }
        printf("Result is = %d\n", sum);
    }
}
```

Output:
3
2 10
Result is = 2047
5 10
Result is = 12207031
10 5
Result is = 111111

/*result ভেরিয়েবল এর ডাটা টাইপ long long ব্যবহার করলেও হবে। long long ডাটা টাইপের ফরমেট স্পেসিফায়ার হলো %lld */

সমস্যা ৪১ – ধারার যোগফল-২

$$\frac{1}{1!} + \frac{2}{2!} + \dots + \frac{n}{n!}$$

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n;
        scanf("%d", &n);
        double sum = 0;
        for (int i = 1; i <= n; i++)
        {
            int factorial = 1;
            for (int j = 1; j <= i; j++) {
                factorial = factorial * j;
            }
            sum = sum + (double)i / factorial;
        }
        printf("%.4lf\n", sum);
    }
}
```

Output:

```
3
5
2.7083
8
2.7183
10
2.7183
```

/*এইভাবে করলেও হবে তবে ফাংশন ইউজ করে করাটা বেশি যুক্তিপূর্ণ এবং ফেক্সিবল।*/

সমস্যা ৪১ – ধারার যোগফল-২

```
#include <stdio.h>
long long fact(int n)
{
    long long sum = 1;
    int i;
    for (i = 1; i <= n; i++)
    {
        sum *= i;
    }
    return sum;
}

int main()
{
    int t, i;
    scanf("%d", &t);
    for (i = 1; i <= t; i++)
    {
        int number;
        scanf("%d", &number);
        int j = 1;
        double sum = 0;

        while (j <= number)
        {
            sum += (double)j / fact(j);
            j++;
        }
        printf("%0.4lf\n", sum);
    }

    return 0;
}
```

Output:	
3	
5	15! বেশ বড় সংখ্যা তাই
2.7083	long long int ব্যবহার করা
8	হয়েছে।
2.7183	এর format specifier হলো
10	%lld
2.7183	

সমস্যা ৪২ – ধারার যোগফল-৩

```
#include <stdio.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int n;
        scanf("%d", &n);
        for (int i = n; i >= 0; i--)
        {
            if (i == 1) printf("2 + ");
            else if (i == 0) printf("1\n");
            else printf("2^%d + ", i);
        }
    }
}
```

Output:

```
3
5
2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1
8
2 ^ 8 + 2 ^ 7 + 2 ^ 6 + 2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1
10
2 ^ 10 + 2 ^ 9 + 2 ^ 8 + 2 ^ 7 + 2 ^ 6 + 2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1
```


সমস্যা ৪৩ – হিসাবকিতাব

/* $p^q \bmod c$ এর মান নির্ণয় করার প্রোগ্রাম লিখতে হবে।*/

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int p, q, c;
        scanf("%d %d %d", &p, &q, &c);

        int sum = 1;
        for (int i = 1; i <= q; i++)
        {
            sum = (sum * p) % c;    /*ডিরেক্ট করে ফেললাম*/
        }
        /*int result = sum % c;*/
        printf("Result = %d\n", sum);
    }
}
```

Output:
3
2 3 3
Result = 2
2 10 5
Result = 4
50 2 3
Result = 1

```
int power = pow(p, q);
int a = power % c;
printf("%d", a);
```

এইভাবে করলেও হবে। উপরে অবশ্যই math.h লিখতে হবে।

সমস্যা ৪৪ – প্যাসকেলের ত্রিভুজ-১

```
#include <stdio.h>
```

```
int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        int row;
        scanf("%d", &row);

        for (int i = 1; i <= row; i++)
        {
            int a = 1;
            for (int j = 1; j <= i; j++)
            {
                printf("%d ", a);
                a = a * (i - j) / j;
            }
            printf("\n");
        }
        printf("\n");
    }
}
```

Output:

```
2
3
1
1 1
1 2 1

4
1
1 1
1 2 1
1 3 3 1
```

3 এর জন্য 3 টা রো এর পরিবর্তে যদি 4 টা রো দিতে চাই তখন শুধু $i \leq \text{row} + 1$ করে দিলেই হবে।

সমস্যা ৪৫ – প্যাসকেলের ত্রিভুজ-২

```
#include <stdio.h>

int main()
{
    int row;
    scanf("%d", &row);
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j <= row - i; j++)
        {
            printf("  "); /*2 space*/
        }
        int a = 1;
        for (int k = 0; k <= i; k++)
        {
            if (i == 0 || k == 0) a = 1;
            else a = a * (i - k + 1) / k;

            printf("%4d", a);
        }
        printf("\n");
    }
}
```

Output:
8

```

      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
```

সমস্যা ৪৬ – ত্রিভুজের ক্ষেত্রফল

```
#include <stdio.h>
#include <math.h>

int main()
{
    int t;
    scanf("%d", &t);
    for (int h = 1; h <= t; h++)
    {
        double a, b, c, s, area;
        scanf("%lf %lf %lf", &a, &b, &c);
        s = (a + b + c) / 2;
        area = sqrt(s * (s - a) * (s - b) * (s - c));
        printf("Area = %.3lf\n", area);
    }
}
```

Output:

```
3
24 30 18
Area = 216.000
13 18 15
Area = 95.917
20 20 20
Area = 173.205
```

ত্রিভুজের তিন বাহুর দৈর্ঘ্য দেওয়া থাকলে ক্ষেত্রফল নির্ণয় করার সূত্র হলোঃ

অর্ধপরিসীমা, $s = \frac{(a+b+c)}{2}$

$\text{Area} = \sqrt{s \times (s-a) \times (s-b) \times (s-c)}$

সমস্যা ৪৭ – আরের জোট

```
#include <stdio.h>

int main()
{
    int f, i;
    scanf("%d", &f);
    int array1[101];
    for (i = 0; i < f; i++)
    {
        scanf("%d", &array1[i]);
    }
    int s, j;
    scanf("%d", &s);
    int array2[101];
    for (j = 0; j < s; j++)
    {
        scanf("%d", &array2[j]);
    }
    int l;
    int add[101];
    for (i = 0, j = 0, l = 0; i < f && j < s; l++)
    {
        if (array1[i] > array2[j])
        {
            add[l] = array2[j];
            j++;
        }
        else
        {
            add[l] = array1[i];
            i++;
        }
    }
    while (i < f)
    {
        add[l] = array1[i];
        l++; i++;
    }
    while (j < s)
    {
        add[l] = array2[j];
        l++; j++;
    }
    for (int a = 0; a < f + s; a++) {
        printf("%d ", add[a]);
    }
}
```

Output:

```
7
10 20 30 40 50 60 70
5
11 22 33 44 55
10 11 20 22 30 33 40 44 50 55 60 70
```

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি
হাবিজিবি হয়ে যাবে বলে আর দিলাম না*/

সমস্যা ৪৮ – নিখোঁজ সংখ্যা

```
#include <stdio.h>
```

```
int main()
{
    int n;
    scanf("%d", &n);
    int array[101];
    for (int i = 0; i < n - 1; i++)
    {
        scanf("%d", &array[i]);
    }
    for (int a = 1; a <= n; a++)
    {
        int count = 0;
        for (int i = 0; i < n - 1; i++) {
            if (a == array[i])
                count = 1;
        }
        if (count == 0) {
            printf("Missing number is = %d\n", a);
            break;
        }
    }
}
```

Output:

7

2 1 4 6 5 3

Missing number is = 7

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজবিবি হয়ে যাবে বলে আর দিলাম না*/

সমস্যা ৪৯ – মৌলিক কি না

```
#include <stdio.h>

int main()
{
    int n;
    scanf("%d", &n);
    int count = 0;
    if (n == 2)
        printf("It's a prime number\n");
    else
    {
        for (int i = 2; i <= n; i++) {
            if (n % i == 0) {
                count++;
                break;
            }
        }
        if (count == 0)
            printf("This is prime number\n");
        else
            printf("This is not a prime number\n");
    }
}
```

Output:
7
This is prime number

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজাবি হয়ে যাবে বলে আর দিলাম না*/

সমস্যা ৫০ – লেফট-রাইট

```
#include <stdio.h>

int main()
{
    char str[51];
    gets(str);
    int length = strlen(str);
    for (int i = 0; i < length; i++)
    {
        if (str[i] == 'L')
            str[i] = str[i - 1];
        else if (str[i] == 'R')
            str[i] = str[i + 1];
    }
    printf("%s\n", str);
}
```

Output:
oL7
oo7

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজবি হয়ে যাবে বলে আর দিলাম না অবশ্যই getchar() ইউজ করতে হবে */

সমস্যা ৫১ – খোঁজ দ্যা সাচ-১

/*দুটি স্ট্রিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্ট্রিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্ট্রিংটি প্রথম কোথা হতে শুরু হয়েছে।*/

```
#include <stdio.h>
#include <string.h>

int main()
{
    char str1[131];
    char str2[131];
    scanf("%s", &str1);
    scanf("%s", &str2);

    int length1 = strlen(str1);
    int length2 = strlen(str2);

    for (int i = 0; i <= length1 - length2; i++)
    {
        if (strncmp(str1 + i, str2, length2) == 0) {
            printf("%d\n", i);
            break;
        }
    }
}
```

Output:
banana ana
1
foorbar foorbar
0

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজবিবি হয়ে যাবে বলে আর দিলাম না।

অবশ্যই getchar() ইউজ করতে হবে*/

সমস্যা ৫২ – খোঁজ দ্যা সাচ-২

/*দুটি স্ট্রিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্ট্রিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্ট্রিংটি মোট কত বার আছে।*/

```
#include <stdio.h>
#include <string.h>

int main()
{
    char str1[131];
    char str2[131];
    scanf("%s", &str1);
    scanf("%s", &str2);

    int length1 = strlen(str1);
    int length2 = strlen(str2);

    int count = 0;
    for (int i = 0; i <= length1-length2; i++)
    {
        if (strncmp(str1 + i, str2, length2) == 0) {
            count++;
        }
    }
    printf("%d\n", count);
}
```

Output:
banana ana
2
foorbar foorbar
1

/*টেস্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজবি হয়ে যাবে বলে আর দিলাম না।

অবশ্যই getchar() ইউজ করতে হবে*/

সমস্যা-৫১ এবং ৫২ বিকল্প নিয়ম

```
#include <stdio.h>
#include <string.h>

int main()
{
    char str1[131];
    char str2[131];
    scanf("%s", &str1);
    scanf("%s", &str2);

    int length1 = strlen(str1);
    int length2 = strlen(str2);
    int i, j, count=0;

    for (i = 0; i <= length1 - length2; i++)
    {
        if (str2[0] == str1[i])
        {
            for (j = 1; j < length2; j++)
            {
                if (str2[j] != str1[i + j])
                    break;
            }
            if (j == length2)
            {
                printf("%d\n", i);
                break;
                /*count++;*/
            }
        }
    }
    /*printf("%d\n", count);*/
}
```

৫১ এর ক্ষেত্রে printf হবে আর ৫২ এর ক্ষেত্রে count হবে।