

Napište program, ktorý načíta celé číslo N nasledované znakom konca riadku. Potom načíta N reálnych čísel, každé nasledované znakom konca riadku. Výstupom programu sú dva riadky, prvý bude obsahovať text `Minimum:` nasledovaný medzerou a minimom z N načítaných čísel vypísanom na 2 desatinné miesta. Druhý riadok výstupu bude obsahovať text `Maximum:` nasledovaný medzerou a maximom z načítaných čísel vypísanom na 2 desatinné miesta a ukončený znakom konca riadku.

Ukážka vstupu:

```
4
5.5
2.78
15.451
6
```

Výstup pre ukázkový vstup:

```
Minimum: 2.78
Maximum: 15.45
```

```
1 // uloha2-1.c -- Peter Plevko, 6.10.2019 17:35
2
3 #include <stdio.h>
4
5 int main()
6 {
7
8     int i,pocetcisel;
9     double maximum,cislo, minimum;
10    scanf("%d", &pocetcisel);
11
12
13    for (i = 1; i <= pocetcisel; i++)
14    {
15
16        scanf("%lf", &cislo);
17        if (i == 1)
18        {
19            maximum = cislo;
20            minimum = cislo;
21        }
22        if (cislo > maximum) maximum = cislo;
23        if (cislo < minimum) minimum = cislo;
24    }
25    printf("Minimum: %.2lf\n", minimum);
26    printf("Maximum: %.2lf", maximum);
27
28    return 0;
29 }
```

```
1 5
2 -55.5
3 -11.1
4 -22.2
5 -33.3
6 -44.4
```

Napište program, ktorý načíta celé číslo n nasledované znakom konca riadku. Potom načíta postupnosť n celých čísel, každé nasledované znakom konca riadku. Program určí, či načítaná postupnosť čísel je správna. Postupnosť je správna, ak:

- Prvé číslo je z rozsahu $<0, 10>$
- Pre každé i -te číslo ($i > 1$) platí, že nie je väčšie ako dvojnásobok predchádzajúceho ($i-1$)-ho čísla, ani menšie ako polovica predchádzajúceho ($i-1$)-ho čísla.

Ak je postupnosť správna, vypíše program správu Postupnosť je správna a odriadkuje, inak vypíše Postupnosť nie je správna a odriadkuje.

Ukážka vstupu:

3
5
7
9

Výstup pre ukázkový vstup:

Postupnosť je správna

```
1 // uloha2-2.c -- Peter Plevko, 6.10.2019 14:35
2
3 #include <stdio.h>
4
5 int main()
6 {
7
8     double x,cislo,i,zaciatok,zaciatok1;
9     scanf("%lf", &zaciatok1);
10    zaciatok = zaciatok1;
11    for (i = 1; i <= zaciatok1; i++)
12    {
13        scanf("%lf", &cislo);
14        if (cislo <= 2 * zaciatok && cislo >= zaciatok / 2) { x = 0; }
15        else x = 1;
16        zaciatok = cislo;
17    }
18    if (x == 0) { printf("Postupnosť je správna"); }
19    else printf("Postupnosť nie je správna");
20
21
22
23
24    return 0;
25 }
```

1 3
2 5
3 7
4 9

1

Napište program, ktorý načíta číslo N. Ak je $N < 1$ alebo $N > 15$, program vypíše chybovú hlášku číslo nie je z daného intervalu a skončí. V opačnom prípade program vypíše N riadkov, kde každý bude obsahovať číslo riadku, dvojbodku, medzeru a čísla oddelené medzerou. V prvom riadku budú za dvojbodkou vypísané čísla od N po 1. V každom nasledovnom riadku bude vždy o jedno číslo menej. Všetky čísla vypisujte na 2 miesta.

Ukážka vstupu:

4

Výstup pre ukážkový vstup:

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

```
1 // uloha2-3.c -- Peter Plevko, 6.10.2019 14:36
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int x,s,d,i, j, rows;
8
9     scanf("%d",&rows);
10    d = rows;
11    s = 0;
12    x = 1;
13    for(i=rows; i>=1; --i,x++)
14    {
15        d = rows;
16        d = d - s;
17
18        s = s + 1;
19
20        printf("%d", x);
21        printf(": ");
22
23        for(j=1; j<=i; ++j,d--)
24        {
25            printf("%d ",d);
26        }
27
28        printf("\n");
29    }
30
31    return 0;
32 }
33
```

Napište program, ktorý načíta číslo N . Ak je $N < 1$, $N > 15$ alebo je N párne číslo, program vypíše chybu zly vstup a skončí. Ak bude program pokračovať, zo znakov "*" a "-" (medzera) nakreslí rovnoramenný trojuholník s výškou N .

Ukážka vstupu:

5
Výstup pre ukážkový vstup:

```
*----
**---
***--
****_
*****
****_
***--
**---
*----
```

Poznámka: Znak '-' vo výstupe reprezentuje medzeru.

```
1 // uloha2-4.c -- Peter Plevko, 6.10.2019 15:20
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int kokes, k, menic, j, i, pocet;
8
9     scanf("%d", &pocet);
10
11     if (pocet < 1 || pocet > 15 || pocet % 2 == 0) {
12         printf("zly vstup");
13     }
14
15     kokes = 1;
16     menic = 1;
17     for (i = 1; i <= pocet + pocet - 1; i++)
18     {
19         for (j = 1; j <= menic; j++)
20         {
21             printf("*");
22         }
23         for (k = pocet; k > menic; k--)
24         {
25             printf(" ");
26         }
27         kokes = kokes + 1;
28         if (kokes <= pocet) menic = menic + 1;
29         else menic = menic - 1;
30         printf("\n");
31     }
32
33
34
35
36
37
```

```

* _ _ * _ _ *
_ * _ _ * _ _
_ _ * _ _ * _ _
_ _ _ * * _ _ _
* * * * *
_ _ _ * * _ _ _
_ _ * _ _ * _ _
_ * _ _ _ * _ _
* _ _ _ _ *

```

```
1 // uloha2-5.c -- Peter Plevko, 8.10.2019 15:40
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int pocethviezd, i, j;
8     scanf("%d", &pocethviezd);
9
10    for (j = 0; j < pocethviezd; j++)
11    {
12        for (i = 0; i < pocethviezd; i++)
13        {
14            if ((i == pocethviezd/2) || (j == pocethviezd/2) || (i == j) || (i == pocethviezd-j))
15                printf("*");
16            else
17                printf("-");
18        }
19        printf("\n");
20    }
21    return 0;
22 }
```

1	3
---	---

Štandardný výstup

1

Kompilácia

Napište program, ktorý načíta v prvom riadku celé číslo N. Potom zo vstupu načíta N riadkov, každý obsahuje jedno celé číslo. Výstupom programu je jeden riadok obsahujúci počet načítaných čísel, ktoré patria do intervalu (0,100>. Výstup je ukončený znakom konca riadku.

Ukážka vstupu:

```
3
5
1000
20
```

Výstup pre ukázkový vstup:

```
2
```

Poznámka: Zamyslite sa ako by to bolo, ak by vopred nebol určený počet čísel na vstupe (číslo N), a mali by ste spracovať všetky čísla. Odpoveď: Treba si všímať návratovú hodnotu funkcie `scanf`.

```
1 // uloha2-6.c -- Peter Plevko, 8.10.2019 16:12
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int x = 0, porovnavac, i, cislo;
8     scanf("%d", &cislo);
9     for (i = 1; i <= cislo; i++)
10     {scanf("%d", &porovnavac);
11      if (porovnavac > 0 && porovnavac <= 100) x = x + 1;
12     }
13     printf("%d", x);
14
15
16
17     return 0;
18 }
```

```
1 9
2 5
3 1000
4 20
5 5
6 1000
7 20
8 5
9 1000
10 20
```

Napište program, ktorý načíta nezáporné celé číslo N.
Výstupom programu je faktoriál čísla N.

Ukážka vstupu:

5

Výstup pre ukážkový vstup:

120

```
1 // uloha2-7.c -- Peter Plevko, 8.10.2019 16:57
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int faktorial,i,cislo;
8     scanf("%d",&cislo);
9     faktorial=1;
10    for (i=1;i<=cislo;i++)
11    {
12        faktorial=faktorial*i ;
13    }
14    printf("%d",faktorial);
15    return 0;
16
17 }
```

1 7

1

Napište program, ktorý načíta dve celé čísla f, g oddelené medzerou. Vstup je ukončený znakom konca riadku. Výstupom je riadok obsahujúci všetky celé čísla deliteľné tromi a patriace intervalu $\langle f, g \rangle$.

Čísla na výstupe sú usporiadané od najmenšieho po najväčšie a sú oddelené vždy jednou medzerou. Výstup je ukončený koncom riadku.

Ukážka vstupu:

5 20

Výstup pre ukázkový vstup:

6 9 12 15 18

```
1 // uloha2-8.c -- Peter Plevko, 8.10.2019 17:32
2
3 #include <stdio.h>
4
5 int main()
6 {
7     int i, a, b;
8     scanf("%d %d", &a, &b);
9     for (i = a; i <= b; i++)
10     {
11         if (i % 3 == 0) printf("%d ", i);
12     }
13
14     return 0;
15 }
```

1 1 2

1

Napište program na vykreslenie šachovnice z hviezdíček (znak '*') a pomlčiek (znak '-'). Šachovnica má rozmer $N \times N$ a jedno políčko tvorí $P \times P$ hviezdíček resp. medzier. Vstupom programu sú dve celé čísla N a P oddelené jednou medzerou. Predpokladajte, že $1 \leq N \leq 8$, a $1 \leq P \leq 4$.

Ukážka vstupu:

2 3

Výstup pre ukážkový vstup:

```
***--  
***--  
***--  
-----  
-----  
-----
```

```
1 // uloha2-9.c -- Peter Plevko, 8.10.2019 19:17  
2  
3 #include <stdio.h>  
4 int main()  
5 {  
6     int n,p,a,b,c,d;  
7     scanf ("%d %d",&n,&p);  
8  
9     for (a=0;a<n;a++)  
10    {  
11        for (b=0;b<p;b++)  
12        {  
13            for(c=0;c<n;c++)  
14            for (d=0;d<p;d++)  
15            if ((a%2==0&&(c%2)==0)||(!(a%2==0&&(c%2)==0))  
16            printf("*");  
17            else printf("-");  
18            printf("\n");  
19        }  
20    }  
21  
22    return 0;  
23 }
```