

Napišimo program, ki ustvari in prikaže vse večkratnike števila 4 med številoma 20 in 60. V izpisu naj bo v vsaki vrsti po 5 števil.

Jezik C#

```
static void Main(string[] args)
{
    for (int vck = 20; vck <= 60; vck = vck + 4)
    {
        if (vck % 5 == 0)                //če je števec zanke deljiv s 5,...
        {
            Console.WriteLine();        //...skočimo v novo vrsto
        }
        Console.Write(vck + " ");      //izpis večkratnika
    }
    Console.ReadLine();
}
```

Jezik C++

```
int main()
{
    int x;
    for (int vck = 20; vck <= 60; vck = vck + 4)
    {
        if (vck % 5 == 0)                //če je števec zanke deljiv s 5,...
        {
            cout << endl;                //...skočimo v novo vrsto
        }
        cout << vck << " ";            //izpis večkratnika
    }
    cin >> x;
    return 0;
}
```

Jezik Java

```
public static void main(String args[])
{
    for (int vck= 20; vck <= 60; vck= vck + 4)
    {
        if (vck % 5 == 0)                //če je števec deljiv s 5...
        {
            System.out.println();        //...skočimo v novo vrsto
        }
        System.out.print(vck + " ");    //izpis večkratnika
    }
}
```

Števec, ki smo ga ustvarili v glavi zanke, je hkrati tudi večkratnik števila 4, saj je korak povečevanja števca 4. Začetno vrednost smo nastavili na spodnjo mejo željenega intervala, zgornjo mejo pa smo določili v pogoju za končanje zanke.

Pogoju za pet števil v vrsti smo zadostili z gnezdenim pogojnim stavkom, v katerega smo postavili ukaz za skok v novo vrsto. Skok se je zgodil vedno, ko je bil števec večkratnik števila 5 (pogoj: ostanek po deljenju s 5 je enak 0).

Napišimo program, ki v konzolo izriše lik iz zvezdic:

```
*
**
***
****
*****
*****
*****
```

Jezik C#

```
static void Main(string[] args)
{
    for(int a = 1; a <= 7; a++)           //sedemkratna ponovitev...
    {
        for (int b = 0; b < a; b++)       //...risanja vrstice zvezdic
        {
            Console.Write("*");
        }
        Console.WriteLine();             //skok v novo vrsto
    }
    Console.ReadLine();
}
```

Jezik C++

```
int main()
{
    int x;
    for(int a = 1; a <= 7; a++)           //sedemkratna ponovitev...
    {
        for (int b = 0; b < a; b++)       //...risanja vrstice zvezdic
        {
            cout << "*";
        }
        cout << endl;                     //skok v novo vrsto
    }
    cin >> x;
    return 0;
}
```

Jezik Java

```
public static void main(String args[])
{
    for (int a= 1; a <= 7; a++)           //sedemkratna ponovitev...
    {
        for (int b= 1; b <= a; b++)     //...risanja vrstice zvezdic
        {
            System.out.print("*");
        }
        System.out.println();           //skok v novo vrsto
    }
}
```

Zunanja zanka določa število vrstic, hkrati pa njen števec določa tudi pogoj v gnezdeni zanki, od katerega je odvisna dolžina vrstice. Ker je števec a z vsako ponovitvijo zanke za 1 večji, je vrstica vsakič daljša za eno zvezdico.