

Ciklai yra naudojami  
pasikartojantiems veiksmams atlikti

**while (sąlyga)**

Sakinys;

**Veiksmai kartojami  
kol tenkinama  
sąlyga**

**while (sąlyga)**

{

Sakinys;

Sakinys;

.....

}

```
a = 10; b = 5
```

```
While (a > b)
```

```
{
```

```
    a = a - 1;
```

```
    b = b + 1;
```

```
    cout << a << b << endl;
```

```
}
```

a	b
9	6
8	7
7	8

```
a = 10; b = 5  
While (a > b)  
{  
    a = a + 1;  
    b = b - 1;  
    cout << a << b << endl;  
}
```

a	b
11	5
12	4
...	...

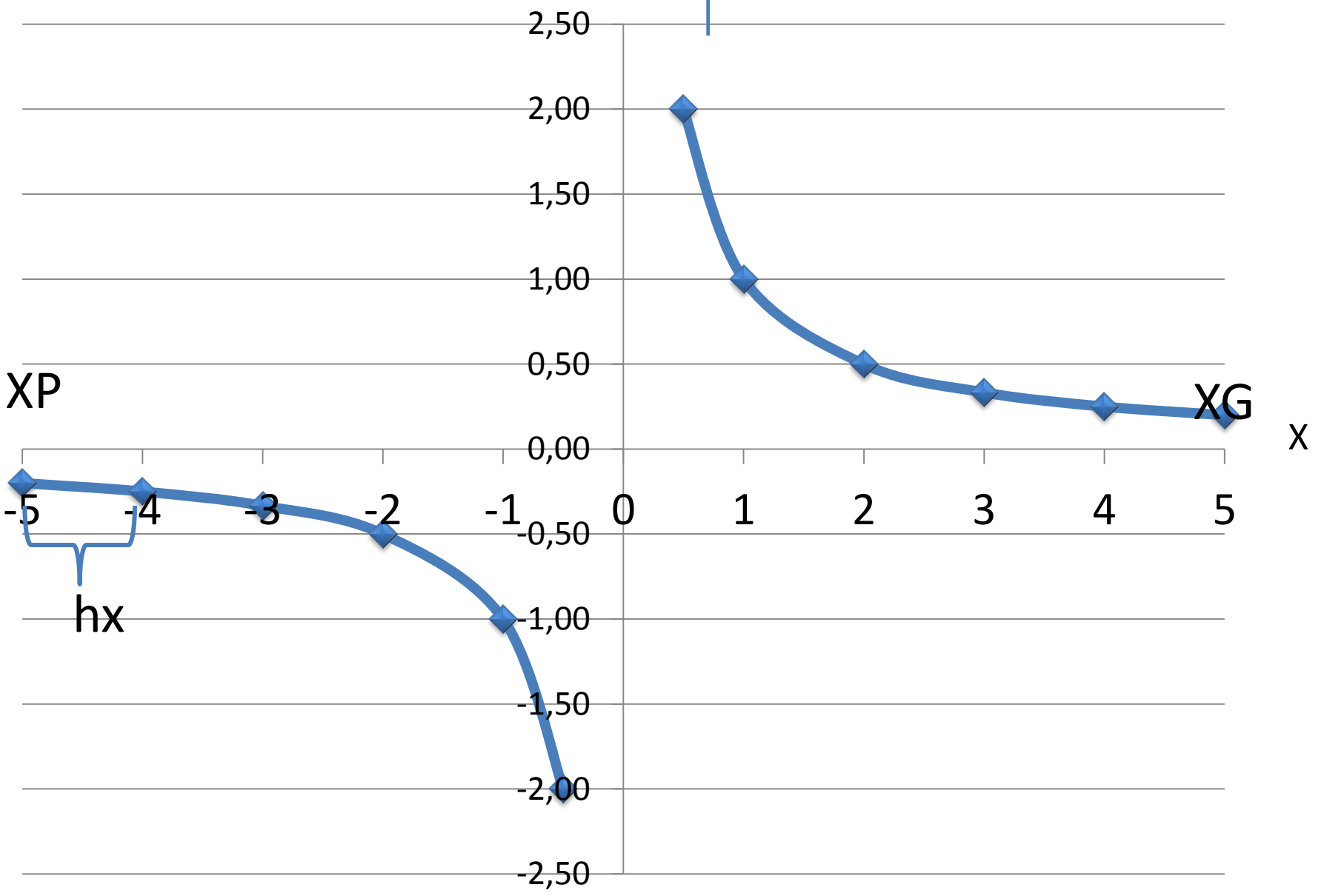
Suskaičiuoti funkcijos  $y = 1/x$  reikšmes, kai  $x$  kinta:  $x_p \leq x \leq x_g$  žingsniu  $hx$ .  
 Suformuoti rezultatų lentelę ir atspausdinti ekrane. Funkcijos reikšmes atspausdinti 2 ženklų po kablelio tikslumu. Duomenys įvedami dialoge klaviatūra. Jei prie tam tikrų argumento reikšmių funkcija neegzistuoja, tai rezultatų lentelėje rašomas žodis „Nėra“.

Rezultatų lentelės pvz:

```

*****
|   x   |   y   |
*****
|       |       |
|       |       |
|       |       |
|       |       |
  
```

$y = 1/x$



XP

XG

hx

x

y

X	Y
-5	-0,20
-4	-0,25
-3	-0,33
-2	-0,50
-1	-1,00
0	Nèra
1	1,00
2	0,50
3	0,33
4	0,25
5	0,20

$$XP = -5$$

$$XG = 5$$

$$hx = 1$$

Īvesti  $XP, XG, hx$

Spausdinti lentelēs antraštē

$X = XP$

kol  $X \leq XG$

	$x \neq 0 ?$
Taip	Ne
$Y = 1/X;$	Sapausdinti $X,$ "Nēra"
Spausdinti $X, Y$	
$X = X + hx;$	

Pabaiga.

```
//funcijos  $y = 1/x$  reiksmiu  
skaiciavimas ir spausdinimas
```

```
#include <iostream>
```

```
#include <cmath>
```

```
#include <fstream>
```

```
#include <iomanip>
```

```
using namespace std;
```

```
int main() {
```

```
float XP, XG, hx, X, Y;
```



```
cout << "Iveskite XP:"; cin >> XP;
cout << "Iveskite XG:"; cin >> XG;
cout << "Iveskite hx:"; cin >> hx;

cout << "*****" << endl;
cout << "|   x   | |   Y   |" << endl;
cout << "*****" << endl;
X = XP;
cout << fixed << setprecision(2);
```

```
while (X<= XG) {
    if (X != 0) {
        Y = 1 / X;
        cout << "|" << setw(7) << X << "|"
<< setw(7) << Y << "|" << endl;
    }
    else cout << "|" << setw(7) << X <<
|" << setw(7) << "Nera" << "|" <<
endl;
    X = X + hx;
}
cout << "*****" << endl;
return 0;}
```