

7. Programování součtů a součinů

Pomocí cyklů realizujeme výpočty matematických výrazů, ve kterých se vyskytují součty a součiny. Součet resp. součin lze obecně popsat schématem $s = \sum_{i=1}^n \boxed{\text{výraz}}$ resp.

$s = \prod_{i=1}^n \boxed{\text{výraz}}$. Odpovídající programová schémata jsou

součet

s:=0;

for i:=1 **to** n **do** s:=s+ $\boxed{\text{výraz}}$

součin

s:=1;

for i:=1 **to** n **do** s:=s* $\boxed{\text{výraz}}$

Příklad 1: Naprogramujme výpočet hodnot výrazů

a)

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots = \sum_{n=0}^k (-1)^n \frac{1}{2n+1}$$

b)

$$1 - \frac{1}{4} + \frac{1}{9} - \frac{1}{16} + \dots = \sum_{n=1}^k (-1)^{n-1} \frac{1}{n^2}$$

c)

$$1 - \frac{1}{27} + \frac{1}{125} - \frac{1}{343} + \dots = \sum_{n=1}^k (-1)^{n-1} \frac{1}{(2n-1)^3}$$

$$d) \frac{4}{5} - \frac{1}{239} - \frac{1}{3} \left(\frac{4}{5^3} - \frac{1}{239^3} \right) + \frac{1}{5} \left(\frac{4}{5^5} - \frac{1}{239^5} \right) - \frac{1}{7} \left(\frac{4}{5^7} - \frac{1}{239^7} \right) + \dots = \sum_{n=0}^k \frac{(-1)^n}{2n+1} \left(\frac{4}{5^{2n+1}} - \frac{1}{239^{2n+1}} \right)$$

procedure TForm1.Sumy(Sender: TObject);

var k,n, {indexy}
 ErrorReport {chybové hlášení} :Integer;
 Jmen1,Jmen2, {jmenovatelé zlomků pro d)}
 Suma {součet} :Extended;
 PrevodRetezec {komunikace s editovacími okny} :String;

begin

Val(EditK.Text,k,ErrorReport);{načtení zadání}

Suma:=0; {případ a)}

For n:=0 **to** k **do**

if odd(n) **then** Suma:=Suma-1/(2*n+1)

else Suma:=Suma+1/(2*n+1);

Str(Suma:18:17,PrevodRetezec);Edit1.Text:=PrevodRetezec;

Suma:=0; {případ b)}

for n:=1 **to** k **do**

if odd(n) **then** Suma:=Suma+1/(n*n)

else Suma:=Suma-1/(n*n);

Str(Suma:18:17,PrevodRetezec);Edit2.Text:=PrevodRetezec;

Suma:=0; {případ c)}

for n:=1 **to** k **do**

if odd(n) **then** Suma:=Suma+1/(2*n-1)/(2*n-1)/(2*n-1)

else Suma:=Suma-1/(2*n-1)/(2*n-1)/(2*n-1);

Str(Suma:18:17,PrevodRetezec);Edit3.Text:=PrevodRetezec;

```

Jmen1:=5;Jmen2:=239;
Suma:=4/Jmen1-1/Jmen2;
for n:=2 to k do
begin
  Jmen1:=Jmen1*5*5;Jmen2:=Jmen2*239*239;
  if odd(n) then Suma:=Suma+1/(2*n-1)*(4/Jmen1-1/Jmen2)
    else Suma:=Suma-1/(2*n-1)*(4/Jmen1-1/Jmen2);
end;
Str(Suma:18:17,PrevodRetezec);Edit4.Text:=PrevodRetezec;
end;

```

{případ d)}

Součty řad

k= 10

Výpočet Konec

$s = \sum_{n=1}^k (-1)^{n-1} \frac{1}{n^2} =$	0.80807895235139817	$4s =$	3.23231580940559269
$s = \sum_{n=0}^k (-1)^n \frac{1}{2n+1} =$	0.81796217561098513	$\sqrt{12}s =$	3.13297719546948213
$s = \sum_{n=1}^k (-1)^{n-1} \frac{1}{(2n-1)^3} =$	0.96888455599560624	$\sqrt[3]{32}s =$	3.14152608792950606
$\sum_{n=0}^k \frac{(-1)^n}{2n+1} \left(\frac{4}{5^{2n+1}} - \frac{1}{239^{2n+1}} \right) =$	0.78539816339744792	$4s =$	3.14159265358979170
		$\pi =$	3.14159265358979324

Na připojeném obrázku vidíme výstup z tohoto programu, k němuž je třeba připojit několik poznámek. Především vzorce na formuláři již nejsou sestavovány z objektů typu TLabel (tak, jako v př. 1 kpt 6). Bylo by to sice možné, ale již velice pracné a výsledek by navíc nevypadal hezky. Vzorce byly editovány v matematickém editoru Math Type a pomocí Clip Boardu uloženy jako obrázky. V Delphi byly použity objekty typu TImage (obrázky) v liště Additional. Do těchto obrázků byly vzorce nahrány použitím vlastnosti Picture objektu Image. Vlastnost Transparent objektu Image1 je třeba nastavit na True. Druhý sloupec výsledků vznikl přidáním naznačených výpočtů k příslušným sekvencím programu a konečně vytištěním čísla π tak, jak ho poskytuje pascalovský typ extended. Je zřejmé, že všechny uvedené řady mohou sloužit k výpočtu tohoto čísla. Při jejich konstrukci je různým způsobem použita Taylorova resp Fourierova řada funkce $\arctg x$. Je zřejmé, že řady konvergují různě rychle. K dosažení přesnosti na dvě desetinná místa musíme v první řadě sečíst asi 300 členů. Použijeme-li řadu d), pak sečtením deseti členů dostaneme 14 desetinných míst.

Poznámka: Fakt, že délka kružnice je přímo úměrná jejímu průměru, byl znám již ve starověku. Konstantu této úměrnosti (dnes označovanou π) na dvě desetinná místa znal již Archimédés (* asi 287 př. n. l., † 212 př. n. l.). Číslo je nazváno po **Ludolphu van Ceulenovi** (*1540, † 1610), profesorovi matematiky na univerzitě v Leidenu, který toto číslo vypočetl na 35 desetinných míst. Svého výsledku si cenil natolik, že si ho nechal vytesat na náhrobek.

Současný "světový rekord" údajně drží Jasamusa Kanada z Tokijské university, který toto číslo spočítal na více než 206 miliard desetinných míst.

Řady pro Ludolphovo číslo:

[Zdrojový kód](#)

[Spustitelný kód](#)

Příklad 2: Naprogramujeme výpočet hodnoty výrazu
$$\frac{1 \cdot 3 \cdot 5 \cdot \dots}{2 \cdot 4 \cdot 6 \cdot \dots} = \frac{\prod_{n=1}^k (2n-1)}{\prod_{n=1}^k (2n)}.$$

Výraz naprogramujeme jako funkci s celočíselným argumentem k, výstupní hodnota musí být samozřejmě reálná:

```
Function Vyras(k:Integer):Double;  
var n :Integer;  
    Citatel, Jmenovatel :LongInt;  
begin  
    Citatel:=1;Jmenovatel:=1;  
    for n:=1 to k do Citatel:=Citatel*(2*n-1);  
    for n:=1 to k do Jmenovatel:=Jmenovatel*2*n;  
    Vyras:=Citatel/Jmenovatel;  
end;
```

Z takto naprogramované funkce je vzoreček „dobře čitelný“, nicméně tento přístup není nejlepší, proto se k této funkci ještě vrátíme.

Komutativita součtu – harmonická řada:

[Zdrojový kód](#)

[Spustitelný kód](#)