

Typ real

```
var cislo : real;
begin
  cislo:= 2.567;
  cislo:= StrToFloat(Edit1.Text);
  Image1.Canvas.TextOut(10,15,FloatToStr(cislo));
end;
```

Naformátovanie výpisu desatinného čísla

Desatinné čísla majú zvyčajne veľký rozsah, ale obmedzenú presnosť

- rozsah typu `real` závisí od platformy,
- presnosť závisí od veľkosti čísla: `Single`, `Double`, `Comp`, `Extended` (zoradené od najmenej presných po najpresnejšie).

Tieto typy čísel majú „plávajúcu“ desatinnú čiarku (číslo je zapísané pomocou $\text{mantisa} \cdot 2^{\text{exponent}}$, kde mantisa aj exponent sú celé čísla, napr. číslo 0,1 má periodický dvojkový zápis $(0,0001100)_2$).

Občas dôjde k nekonečnému desatinnému zápisu hodnoty: $1,25 + 0,25 = 1,49999999999999$, preto hodnoty vieme vypísať na požadovaný (potrebný) počet desatinných miest:

```
Image1.Canvas.TextOut(10, 15, FloatToStrf(názovPremennej,ffnumber,
presnost,početDesatinnýchMiest));
```

Pretypovanie real na integer

Aby sme z reálneho čísla dostali celé číslo, je treba toto konvertovať/zaokrúhliť na celé číslo. Na to slúžia 2 funkcie

trunc – výsledkom je celočíselná časť reálnej hodnoty

round – výsledkom je zaokrúhlená hodnota reálnej hodnoty

```
var a:integer;
    o:real;
o:=5.8;
a:=trunc(o);    //a = 5
a:=round(o);    //a = 6
```

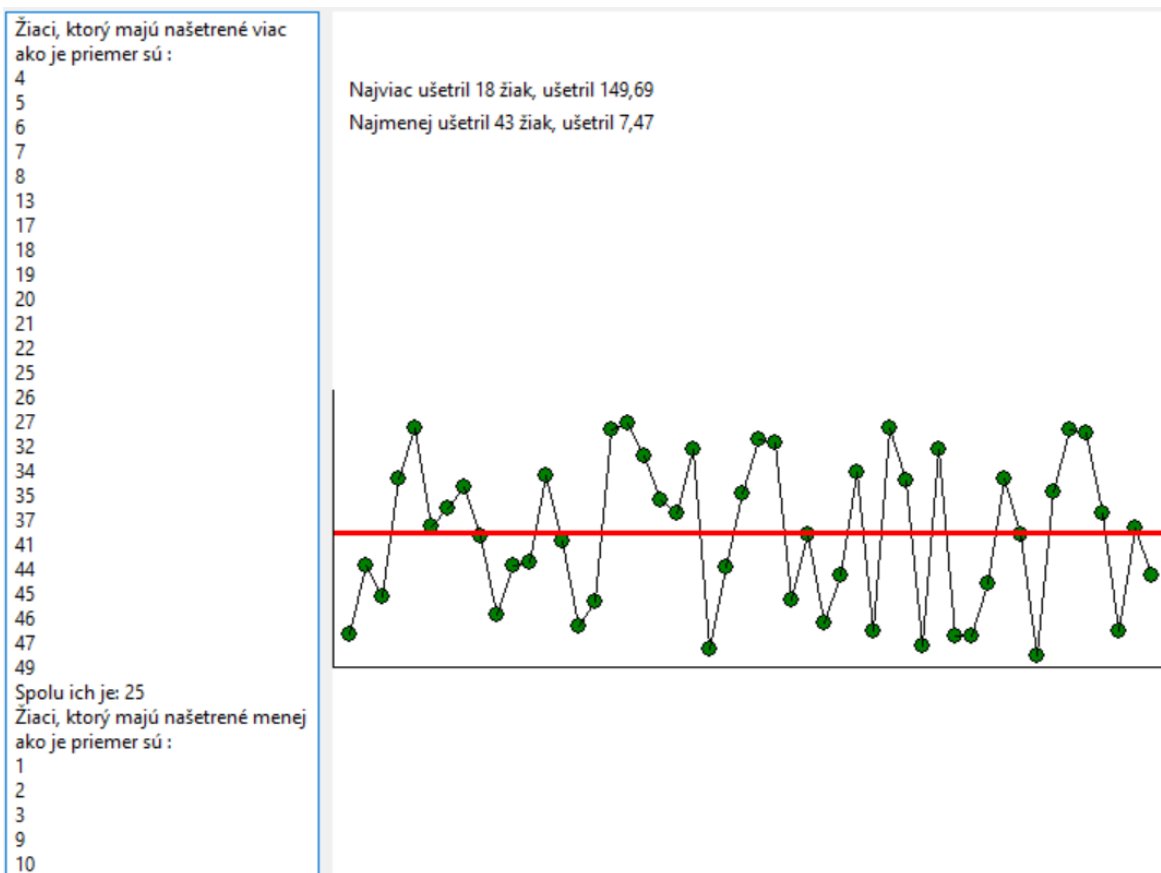
Práca so súborom

Formátovací parameter vo write

- za **znakom** alebo **znakovým reťazcom** `write(t, '*':10);`
 - označuje, že znak sa vypíše na šírku 10, t.j. najprv 9 medzier a potom '*'
 - keď je reťazec dlhší ako formátovací parameter, zapíše sa kompletný reťazec, t.j. ignoruje sa formát
- za **celým číslom** označuje šírku, do ktorej sa má zapísať číslo, ak by nevošlo do danej šírky, formát sa ignoruje
`write(t, 25*25 :5);` zapíše dve medzery, za ktoré dá číslo 625

- za **reálnym číslom** označuje šírku, do ktorej sa má číslo zapísať, desatinné číslo sa vypíše v *semilogaritmickom tvare*; druhý formátovací parameter označuje počet desatinných miest
`write(t, 100/3 :7 :4) ;` zapíše 33.3333

1. Na konci žiackej brigády (3 týždne) sa zamestnávateľ rozhodol odmeniť žiakov a k ich platu (2.5 € za hodinu) im pridal ešte odmenu. Žiakom, ktorí odpracovali viac ako 30 hodín, dal odmenu 10% z ich platu. Ostatní dostali odmenu 5% zo svojho platu. Vytvorte program, ktorému zadáme počet žiakov, vymyslí pre každého žiaka počet odpracovaných hodín, vyplatenú sumu a určí počet žiakov, ktorým bola vyplatená suma aspoň 100 €.
2. Vytvorte 50 prvkové pole, ktoré reprezentuje úspory žiakov jednej triedy (žiaci majú nasparené od 5,00€ do 149,99€ - generovanie náhodných desatinných čísel vytvorte ako generovanie 5ciferných celých čísel, ktoré podelíte 100). Tieto hodnoty reprezentujte čiarovým grafom. Vypočítajte priemernú hodnotu úspor (priemer vykreslite do grafu).
 - a. Do Mema vypíšte, ktorí žiaci (poradové číslo žiaka v poli) majú úspory POD a NAD priemerom. Počet žiakov pod a nad priemerom spočítajte.
 - b. Zistite poradové číslo žiaka, ktorý má nasparené najviac a ktorý má nasparené najmenej, jeho poradové číslo a nasparenú hodnotu vypíšte do Image.



3. V textovom súbore **teploty.txt** sú čísla, každé v jednom riadku, ktoré predstavujú teploty v °C namerané v priebehu 20 dní. Vytvorte program, ktorý prepočíta dané teploty na stupne Fahrenheita ($F = ^\circ C * 9/5 + 32$) a zobrazí získané údaje v tvare:

1. deň 24 °C = 75,2 F

2. deň 15 °C = 59,0 F

.

.

20. deň -5 °C = 23,0 F

Program stupne Fahrenheita zapíše do textového súboru **premenastupnov.txt** na 3 desatinné miesta (každé do jedného riadka), tiež umožní prepočet priemernej dennej teploty v °C a zistí najväčšiu odchýlku od priemeru. Namerané teploty v °C zobrazí v podobe čiarového grafu.