

Riešenie úloh

1. Dá sa jednoducho odvodiť vzorec pre počet zavolaní podprogramu **beep** (je to suma nepárnych čísel od 3 vyššie až po $2n+1$): tento celkový počet je

$$(n+1)^2 - 1$$

- a) pre **N = 5** je to 35 volaní **beep**, teda 3.5 sekundy
b) najmenšie **N**, pre ktoré je počet volaní **beep** aspoň 300 je pre **N = 17** (323 volaní **beep**)
2. a) v osmičkovej sústave $123^2 = 15351$ (zodpovedá to desiatkovému $83^2 = 6889$)
b) najmenšie trojciferné osmičkové číslo **111**, najväčšie **777**, čo je v desiatkovej **73** a **511**
3. postupne vykonáme operácie **ppodpo** šifrovacej postupnosti operácií:

- a) pre slovo 'programovanie':

=p=> 'ramovanieprog'

=p=> 'vanieprogramo'

=o=> 'omargorpeinav'

=d=> 'moraogprieianv'

=p=> 'ogprieianvmora'

=o=> '**aromvnaeirpgo**' zašifrované vstupné slovo

- b) pre slovo 'noyph' vykonáme operácie v opačnom poradí (operáciu **p** aj v opačnom význame):

<=o= 'thpyon'

<=p= 'pyonth'

<=d= 'ypnoht'

<=o= 'thonpy'

<=p= 'onpyth'

<=p= '**python**' odšifrované vstupné slovo

4. Program v Pascale:

```
const n = 5;
var pocet: integer;
    x, sucet, min, max: real;
begin
  read(x);
  pocet := 1;
  sucet := x; min := x; max := x;
  read(x);
  while x >= 0 do
    begin
      if x > max then max := x;
      if x < min then min := y;
      sucet := sucet + x;
      pocet := pocet + 1;
      read(x);
    end;
  writeln('skore = ', (sucet-min-max)/(pocet-2))
end;
```

alebo v Pythone:

```
read(x)
sucet = min = max = x
pocet = 1
read(x)
while x >= 0:
  if x > max: max = x
  if x < min: min = x
  sucet = sucet + x
  pocet = pocet + 1
  read(x)
print('skore =', (sucet-min-max)/(pocet-2))
```