

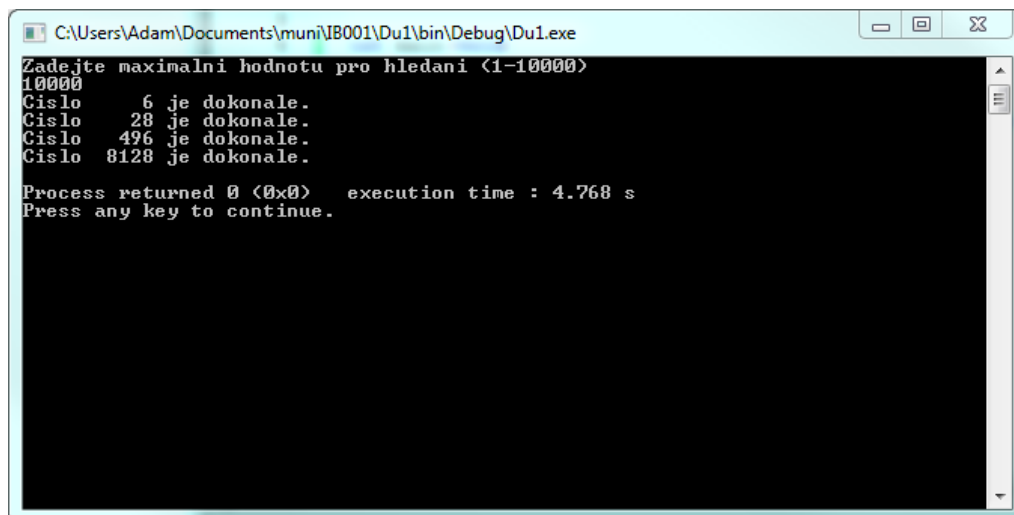
# Hledání dokonalých čísel

Dokonalé číslo je takové, pro které platí, že se rovná součtu všech jeho dělitelů (kromě sebe sama). Např. 6 je dokonalé číslo, protože je dělitelné: 1, 2, 3. 28 je dokonalé číslo, protože je dělitelné: 1, 2, 4, 7, 14.

Program bude fungovat tak, že uživatel zadá číslo a program nalezne všechna dokonalá čísla, která jsou menší nebo rovna tomu zadanému. Program musí dále splňovat následující požadavky (čtete pozorně, jednak za nedodržení budu strhávat body a jednak je tam občas i nějaká nápověda):

- Omezte uživatele na zadávání čísel od 1 do 10000. Pokud uživatel zadá záporné číslo, nulu, nebo příliš velkou hodnotu, vypište chybovou hlášku a ukončete program. (nápověda: program lze kdykoliv ukončit příkazem **return 0;**)
- Komunikujte s uživatelem, řekněte mu, co se po něm chce (že má zadat číslo a jaké hodnoty program akceptuje).
- Čísla ve výstupu zarovnávejte na 5 míst, tak aby číslice na úrovni řádu jednotek byly zarovnané pod sebou bez ohledu na to, kolik číslic číslo má (podívejte se na slide 42 k první přednášce - u příkazu `printf` můžeme specifikovat délku výpisu, ukazoval jsem vám to na prvních nebo druhých cvičeních. Např. pokud v `printf` použijete `%3i`, vypíšete hodnotu proměnné tak, aby vždy zabírala 3 znaky).
- Čísla vypisujte od nejmenšího.
- Vypisujte pouze ta čísla, která jsou dokonalá.
- Program by měl pracovat efektivně, tedy např. pokud testuje číslo 60 a najde dělitele 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, nemusí těch dalších 40 zkoušet, protože už teď je jejich součet více než 60. Dále se zamyslete nad tím, jestli je pro hledání dělitele opravdu třeba testovat všechna čísla menší jak to dělené, nebo to jde ještě nějak více omezit. (Má u čísla 6 opravdu smysl zkoušet, jestli je dělitelná číslem 4?)
- Čísla opravdu hledejte, 4x `printf` s napevno nastavenými hodnotami ze screenshotu nebude považován za správné řešení. :)

Program má zhruba 30 řádků. Ukázka vzorového řešení:



```
C:\Users\Adam\Documents\muni\IB001\Du1\bin\Debug\Du1.exe
Zadejte maximalni hodnotu pro hledani <1-10000>
10000
Cislo      6 je dokonale.
Cislo     28 je dokonale.
Cislo    496 je dokonale.
Cislo   8128 je dokonale.

Process returned 0 (0x0)   execution time : 4.768 s
Press any key to continue.
```

Až budete mít úkol hotový a odevzdaný, můžete zkusit program upravit a spustit tak, aby hledal dokonalá čísla až do 100 000. Proč to najednou trvá neúměrně dlouho, i když se množství čísel zvětšilo jenom 10x? (Pokud chcete čas měřit přesněji, odstraňte zadávání hodnoty uživatelem a nastavte ji napevno do proměnné na 100 000, Code::Blocks vám pak po skončení výpočtu řeknou jeho "čistý čas").

Pokud vám v zadání není něco jasné, napište mi, pokusím se vám to dovysvětlit. Hodně štěstí při programování!