



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Moderní přístup k aplikaci matematických dovedností v přírodovědných a ekonomických oborech
reg. č.: CZ.1.07/2.2.00/28.0168

WolframAlpha ve výuce přírodovědných a ekonomických předmětů

Jan Říha, František Látal, Veronika Říhová

Olomouc 2015

Oponenti: doc. RNDr. Jiří Pechoušek, Ph.D.
RNDr. Vladimíra Mádrová, CSc.

Neoprávněné užití tohoto díla je porušením autorských práv a může zakládat občanskoprávní, správněprávní popř. trestněprávní odpovědnost.

Tato publikace neprošla redakční jazykovou úpravou.

© Jan Říha, 2015

© Univerzita Palackého v Olomouci, 2015

Olomouc 2015

1. vydání

ISBN 978-80-244-4471-0

Obsah

1. Úvod	4
2. WolframAlpha pro fyziky	7
2.1 Astronomická data	7
2.2 Jaderné elektrárny	16
3. WolframAlpha pro chemiky	22
3.1 Chemické prvky a chemické reakce	22
4. WolframAlpha pro geography	26
4.1 Počasí	26
4.2 Statistiky obyvatelstva	27
4.3 Mapy	31
4.4 Tornáda, sopečné erupce a zemětřesení	34
5. WolframAlpha pro matematiky a ekonomy	44
5.1 Bitcoin	44
5.2 Využití WolframAlpha ve finanční matematice	45
5.3 Grafy v kartézské soustavě	55
5.4 Řešení příkladů krok za krokem	60
5.5 Vizualizace rovnic	69
6. Závěr	71
7. Použitá literatura	72

1 Úvod

Každý, kdo pracuje s Internetem, si zvykl používat tzv. vyhledávače, což jsou služby, které umožňují na Internetu najít webové stránky obsahující požadované informace. Alternativu k těmto vyhledávačům přinesla v roce 2009 společnost Wolfram Research, když uvedla svůj “vyhledávač” WolframAlpha. Samotní tvůrci však tuto službu nenazývají pouze “vyhledávačem”, ale tzv. “computational knowledge engine”, což bychom volně mohli přeložit například jako “výpočetní vyhledávací službu”. Slovním zadáním (nikoliv pomocí určité syntaxe) problému v anglickém jazyce obvykle získáme nejen výsledek, ale také mnoho dalších souvisejících informací. Informace jsou navíc členěny do přehledných grafů a tabulek. Nezískáváme tedy pouze odkazy na jiné webové stránky. WolframAlpha nám poskytuje nebo se snaží poskytnout komplexní a přehledně zpracovanou odpověď na námi vznesený dotaz.

Pomocí WolframAlpha můžeme hledat informace ze širokého spektra oborů, jako jsou matematika, fyzika, chemie, biologie, astronomie, historie, kultura, ekonomie, meteorologie a mnoha dalších. Zejména v matematice je WolframAlpha silným nástrojem pro řešení úloh. Najde nám totiž jak výsledek například nějakého neurčitého integrálu, tak také postup, kterým se takový integrál řeší. Vzhledem k tomu, že podporuje také mnoho typů moderních mobilních telefonů, tabletů a čteček, můžeme nainstalováním odpovídající aplikace získat navíc bezkonkurenční kalkulačku.

Cílem naší publikace není obecný návod, jak WolframAlpha používat. WolframAlpha se poměrně rychle vyvíjí a obecný návod zřejmě ani podat nelze. Snažíme se pouze na příkladech z různých oborů ukázat, jaké výsledky WolframAlpha dokáže najít a zobrazit, jak by mělo vypadat zadání hledané úlohy a jakým způsobem by tuto službu mohli využít studenti zejména středních nebo vysokých škol.

Pro zadání dotazů je využito prostředí programu Wolfram *Mathematica* verze 10.0, kde úlohu zapisujeme buď do předvoleného vstupu pro WolframAlpha (WolframAlpha query)



Answer to the Ultimate Question of Life, the Universe, and Everything

Assuming The Ultimate Answer | Use Monty Python's Meaning of Life instead

Input interpretation: +
 Answer to the Ultimate Question of Life, the Universe, and Everything

Result: +
 42
(according to Douglas Adams' humorous science-fiction novel *The Hitchhiker's Guide to the Galaxy*)

WolframAlpha +

nebo pomocí příkazu **WolframAlpha** syntaxe programu Wolfram *Mathematica* (Wolfram Language input).

In[2]= `WolframAlpha["Answer to the Ultimate Question of Life, the Universe, and Everything"]`

Assuming The Ultimate Answer | Use [Monty Python's Meaning of Life](#) instead

Input interpretation: +

Answer to the Ultimate Question of Life, the Universe, and Everything

Result: +

42
(according to Douglas Adams' humorous science-fiction novel *The Hitchhiker's Guide to the Galaxy*)

WolframAlpha +

Další volby příkazu **WolframAlpha** potom slouží zejména ke specifikaci výstupu. Takový výstup by bylo možné získat zadáním příslušné otázky pomocí webové stránky www.wolframalpha.com nebo využitím aplikace v podporovaném mobilním zařízení.

Silnou stránkou WolframAlpha je možnost srovnávání nejrůznějších objektů. Můžeme se tedy pomocí WolframAlpha podívat na srovnání nástroje wolframalpha.com s vyhledávačem google.com.

In[3]= `WolframAlpha["wolframalpha vs. google",
IncludePods -> "HostInformationPod:InternetData", AppearanceElements -> {"Pods"},
TimeConstraint -> {30, Automatic, Automatic, Automatic},
PodStates -> {"HostInformationPod:InternetData__Show map",
"HostInformationPod:InternetData__Hide map",
"WebSiteStatisticsPod:InternetData__Show history",
"WebSiteStatisticsPod:InternetData__Daily visitors history_Last month",
"HostInformationPod:InternetData__More"}]`

Web hosting information: Show map Less

	wolframalpha.com	google.com
name	Wolfram Alpha LLC	Google Inc.
location	Champaign, Illinois, United States	Mountain View, California, United States
coordinates	40° 6' 54" North 88° 16' 25" West	37° 23' 58" North 122° 4' 46" West
city population	82 517 people	76 621 people
local time	3:11 am CST 27. 1. 2015	1:11 am PST 27. 1. 2015
local weather	30 °F no wind N	57 °F no wind N

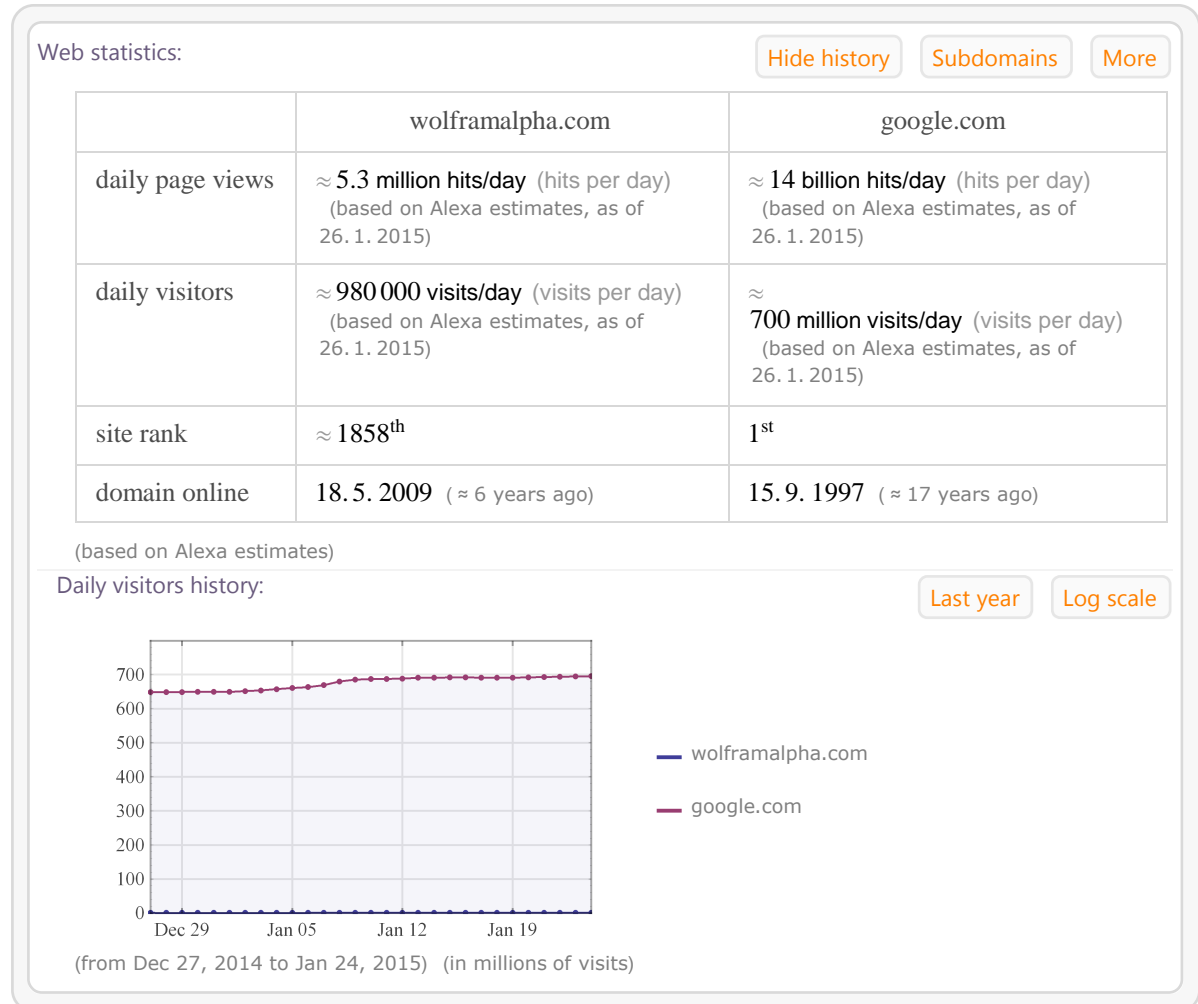
Out[3]=

```

In[4]:= WolframAlpha["wolframalpha vs. google",
  IncludePods -> "WebSiteStatisticsPod:InternetData", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"HostInformationPod:InternetData__Show map",
    "HostInformationPod:InternetData__Hide map",
    "WebSiteStatisticsPod:InternetData__Show history",
    "WebSiteStatisticsPod:InternetData__Daily visitors history_Last month",
    "HostInformationPod:InternetData__More"}]

```

Out[4]=



2 WolframAlpha pro fyziky

2.1 Astronomická data

WolframAlpha obsahuje nepřeberné množství astronomických dat. WolframAlpha umožňuje provádět výpočty, zkoumá vlastnosti a umístění objektů v naší sluneční soustavě. Při dotazu na nějaký astronomický objekt nebo jev na obloze je pozice určena vzhledem k geografické poloze žadatele. WolframAlpha určí tuto polohu na základě IP adresy počítače.

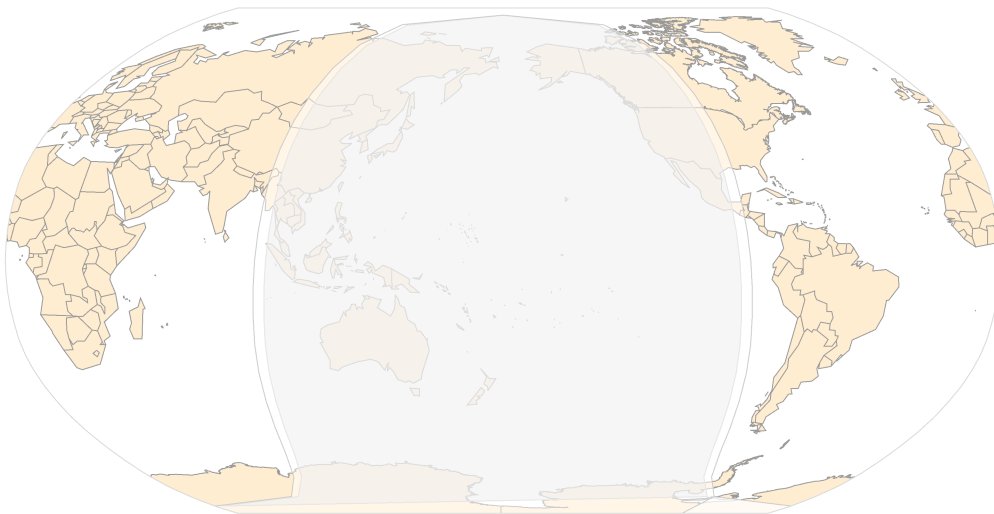
Např. při dotazu na zatmění Měsíce *"lunar eclipse"*, se zobrazí časově nejbližší zatmění Měsíce pozorovatelné na našem území a čas začátku a konce zatmění je uveden v našem středoevropském čase. Vidíme také, zda se jedná o částečné nebo úplné zatmění.

```
In[5]= WolframAlpha["lunar eclipse",
  IncludePods -> {"Result", "EclipseVisibility", "EclipseTimes"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Result:

date	1:59 pm CEST Saturday, April 4, 2015 (67.12 days from now)
type	total

Eclipse visibility:



Eclipse times:

begin partial	12:14 pm CEST Saturday, April 4, 2015
begin total	1:53 pm CEST Saturday, April 4, 2015
end total	2:05 pm CEST Saturday, April 4, 2015
end partial	3:44 pm CEST Saturday, April 4, 2015

WolframAlpha může poskytnout zajímavé informace o vzdálenosti, teplotě a rozměrech objektů ve sluneční soustavě, které

odpovídají denní době a geografické poloze žadatele. Je výhodné, že hodnota je určena pro danou chvíli, kdy se uživatel dotazuje. Naproti tomu v učebnicích se mohou objevovat pouze průměrné hodnoty.

Např. při dotazu na Slunce “Sun” se dovíme východ a západ Slunce na místě, kde byl dotaz položen. Dále se dovíme aktuální polohu na obloze pro daný okamžik dotazu. Další informací je např. aktuální vzdálenost Slunce od Země v astronomických jednotkách, nebo aktuální postavení Země, Slunce a Měsíce. Samozřejmostí jsou také údaje o hmotnosti Slunce, stáří či povrchové teplotě.

```
In[6]:= WolframAlpha["Sun",
  IncludePods -> {"DaylightInformationForDateInLocation", "SunDayPlot:AstronomicalData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[6]=

Daylight information for January

More

27 in Olomouc, Czech Republic:

sunrise	7:32 am CET (2 hours 41 minutes ago)
sunset	4:39 pm CET (6 hours 25 minutes from now)
duration of daylight	9 hours 7 minutes

Sun path today for Olomouc, Olomoucky:

Show zenith

Show DMS

next maximum altitude	21.89° (↙)
next maximum altitude time	12:05 pm CET Tuesday, January 27, 2015
azimuth at rise	117.821° (ESE)
azimuth at set	241.919° (WSW)


```
In[7]:= WolframAlpha["Sun",
  IncludePods -> {"BasicStar:AstronomicalData", "CurrentEarthSunMoonConfiguration"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

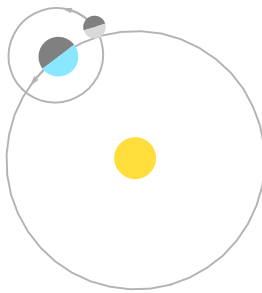
Out[7]=

Star properties: More

distance from Earth	0.9847 au 8.189 light minutes
apparent magnitude	-26.72 (visible to the naked eye in daylight)
absolute magnitude	+4.85 (visual)
spectral class	G2V (main sequence)
effective temperature	5780 K
mass	1.988435×10^{30} kg $332\,948.6 M_{\oplus}$
age	4.57 billion yr
main sequence lifetime	10 billion yr
end state	carbon- oxygen white dwarf

+ Units

Current Earth-Sun-Moon configuration:



(not to scale)

Další informací, kterou si mohou zájemci o astronomii vyhledat na WolframAlpha jsou údaje o planetách pro daný konkrétní den.

Např. zadáním příkazu “*Venus 14 Jan 2001*” vidíme ve WolframAlpha kromě základních obecných údajů o planetě Venuši také vzdálenost Venuše od Země či od Slunce dne 14. ledna 2001. Dále postavení Venuše a planet kolem ní ve sluneční soustavě nebo pozici Venuše na obloze z naší geografické polohy. Zadáním anglických názvů dalších planet sluneční soustavy (např. “*Mercury*”, “*Mars*”, “*Uranus*”) získáme obdobné informace i o těchto planetách.

```
In[8]:= WolframAlpha["Venus 14 Jan 2001",
  IncludePods -> {"BasicPlanetOrbitalPropertiesEntityTriggered:AstronomicalData",
  "BasicPlanetPhysicalProperties:AstronomicalData",
  "PlanetAtmospheres:AstronomicalData"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[8]=

Orbital properties on 12:00 am January 14, 2001:
[Show non-metric](#) [More](#) [Show history](#)

distance from Earth	0.7109 au 5.913 light minutes
average distance from Earth	1.14 au 9.45 light minutes
distance from Sun	0.7212 au 5.998 light minutes
largest distance from orbit center	1.0894185×10^8 km 0.7282313 au
nearest distance from orbit center	1.07476×10^8 km 0.7184327 au
orbital period	224.7008 days

[Units](#)

Physical properties:
[Show non-metric](#) [More](#)

equatorial radius	6051.9 km $0.94885 a_{\oplus}$
mass	4.86732×10^{24} kg $0.814996 M_{\oplus}$
rotation period	243.018 days (sidereal, retrograde)
number of moons	0
age	4.5 billion yr

[Units](#)

Atmosphere:
[Use Fahrenheit](#)

[Show minor constituents](#) [Show pie chart](#)

atmospheric pressure	90 bars (at surface)
average temperature	460 °C (at surface)

Major constituents:

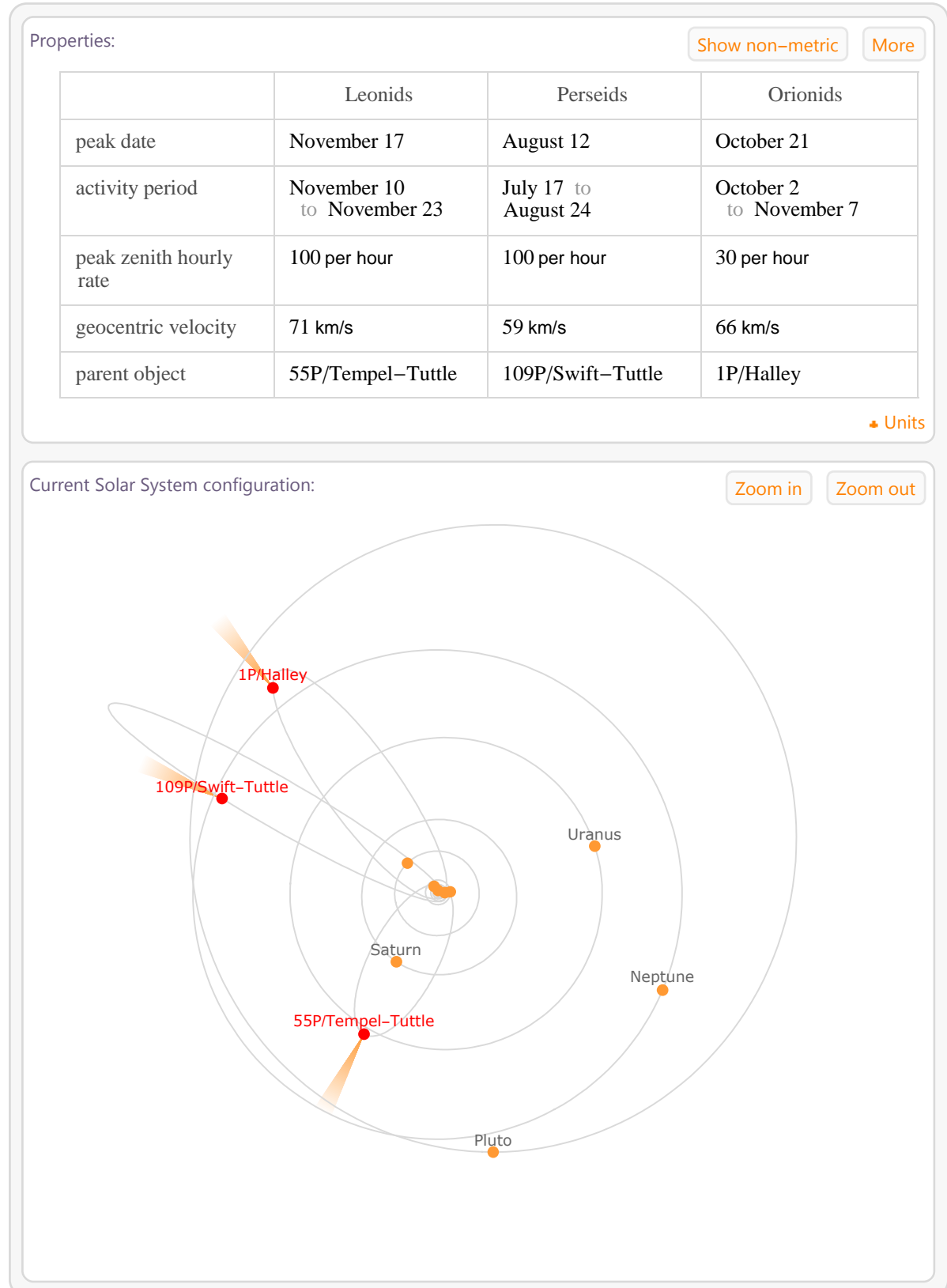
carbon dioxide (CO ₂)	96.5%
nitrogen (N ₂)	3.5%

(components may not add up to 100% due to uncertainty, variability, and round-off)

Jednou z dalších možností, kterou WolframAlpha nabízí je srovnávání více objektů společně. Můžeme se např. podívat na vlastnosti meteorických rojů. Při zadání “*leonids perseids orionids*” vidíme informace o meteorických rojích Leonid, Perseid a Orionid. Lze srovnávat data, kdy je možno tyto roje nejlépe pozorovat. Dále vidíme období výskytu rojů, název a trajektorii

komety, která způsobuje viditelnost těchto rojů. Komety nás zajímají z mnoha důvodů, jednou z nich může být otázka, zda na Zemi přinesla život právě některá z nich.

```
In[9]:= WolframAlpha["leonids perseids orionids",
  IncludePods -> {"Properties:MeteorShower", "SolarSystemConfiguration:MeteorShower"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```



WolframAlpha disponuje značným množstvím informací a údajů, které lze jednoduše srovnávat. Zajímá nás, zda je větší rozloha Ruska nebo povrch trpasličí planety Pluto, stačí napsat "area of Russia, area of Pluto" a máme výsledek. Chceme-li srovnat velikost Země, Marsu a jejich měsíců, napíšeme příkaz "Earth, Mars, Moon, Phobos, Deimos size comparison" a máme vše rychle a názorně k dispozici. Když chceme spočítat poměr hmotnosti hvězdy Proxima Centauri a Slunce stačí zadat příkaz "mass Proxima Centauri / Sun" a okamžitě získáme výsledek.

```
In[10]:= WolframAlpha["area of Russia, area of Pluto", IncludePods -> "Result",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[10]=

Results: Show non-metric

Russia	total area	$1.708 \times 10^7 \text{ km}^2$ (square kilometers) (world rank: 1 st)
Pluto	surface area	$1.7 \times 10^7 \text{ km}^2$ (square kilometers)

```
In[11]:= WolframAlpha["Earth, Mars, Moon, Phobos, Deimos size comparison",
  IncludePods -> {"Sizes:AstronomicalData", "SizeComparison:AstronomicalData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[11]=

Results: Show non-metric

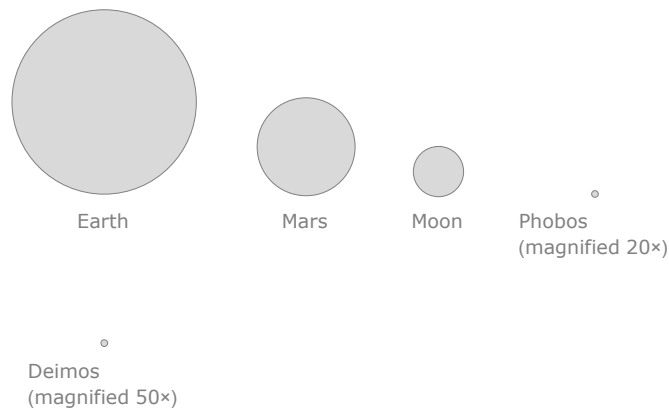
	average radius	equatorial radius	polar radius
Earth	6367.4447 km (kilometers)	6378.137 km (kilometers)	6356.7523142 km (kilometers)
Mars	3386 km (kilometers) $\approx 0.5308 a_{\oplus}$ (equatorial radii of Earth)	3396.19 km (kilometers) $\approx 0.532474 a_{\oplus}$ (equatorial radii of Earth)	3375 km (kilometers) $\approx 0.5292 a_{\oplus}$ (equatorial radii of Earth)
Moon	1737.5 km (kilometers) $\approx 0.27241 a_{\oplus}$ (equatorial radii of Earth)	1738.14 km (kilometers) $\approx 0.272515 a_{\oplus}$ (equatorial radii of Earth)	1735.97 km (kilometers) $\approx 0.272175 a_{\oplus}$ (equatorial radii of Earth)
Phobos	11.1 km (kilometers)		
Deimos	6.2 km (kilometers)		

	average diameter	equatorial diameter	polar diameter
Earth	12 734.889 km (kilometers)	12 756.274 km (kilometers)	12 713.504628 km (kilometers)
Mars	6771 km (kilometers)	6792.38 km (kilometers)	6750 km (kilometers)
Moon	3475 km (kilometers)	3476.28 km (kilometers)	3471.94 km (kilometers)
Phobos	22.2 km (kilometers)		
Deimos	12 km (kilometers)		

	equatorial circumference	angular diameter
Earth	40 007.835 km (kilometers)	180° (degrees)

Mars	21 270 km (kilometers)	4.5" (arc seconds)
Moon	10 917 km (kilometers)	31 arc minutes 30 arc seconds
Phobos	69.7 km (kilometers)	15 mas (milliarc seconds)
Deimos	39 km (kilometers)	8.2 mas (milliarc seconds)

Physical size comparison:



```
In[12]:= WolframAlpha["mass Proxima Centauri / Sun", IncludePods -> {"Input", "Result"},
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
PodStates -> {"Result__Show details"}]
```

Input interpretation:

Proxima Centauri mass
Sun mass

Out[12]=

Results:

[Hide details](#)

0.21 (21%)

Proxima Centauri mass	4.2×10^{29} kg (kilograms)
Sun mass	1.988435×10^{30} kg (kilograms)

S pomocí WolframAlpha lze vypsát seznam 10 nejjasnějších hvězd, které jsou pro nás viditelné. Kromě seznamu hvězd jsou k dispozici podrobné informace, např. aktuální vzdálenost hvězd od Země, jejich barva, velikost, teplota nebo umístění v HR diagramu.

```
In[13]:= WolframAlpha["10 brightest stars",
  IncludePods -> {"Result", "HertzsprungRussellDiagram:AstronomicalData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Result: More

1	Sun	-26.72
2	Sirius	-1.44
3	Canopus	-0.62
4	Arcturus	-0.05
5	Rigel Kentaurus A	-0.01
⋮		

Hertzsprung–Russell diagram:

Out[13]=

WolframAlpha nabízí řadu map objektů sluneční soustavy. Můžeme např. prozkoumat povrch Měsíce, pomocí příkazu “*map of the moon*”, kde WolframAlpha umožňuje i možnost 3D zobrazení povrchu Měsíce.

```
In[14]:= WolframAlpha["map of the moon", IncludePods -> "SurfaceMap:AstronomicalData",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"SurfaceMap:AstronomicalData__Orthographic projection"}]
```

Results:

Cylindrical projection



(Moon centered at 0° E longitude)



(Moon centered at 90° E longitude)



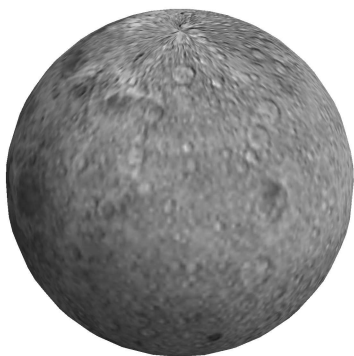
(Moon centered at 180° E longitude)



(Moon centered at 270° E longitude)

Out[14]=

3D image:



Velkou výhodou WolframAlpha, je, že data, která zobrazuje jsou aktuální v okamžiku dotazu. Např. můžeme dotazem *"location of international space station"* zjistit aktuální pozici Mezinárodní vesmírné stanice.

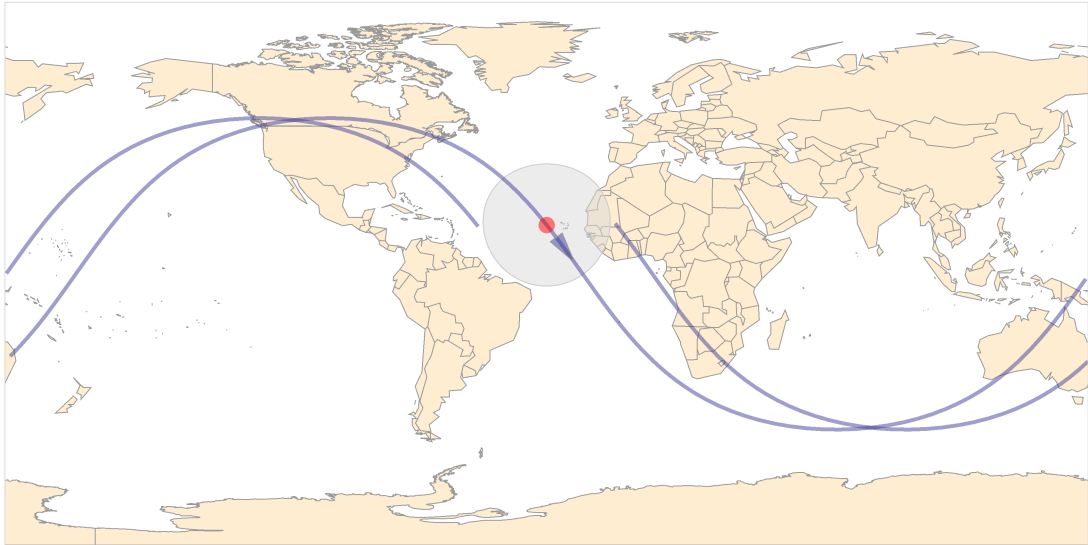
```
In[15]:= WolframAlpha["location of international space station",
  IncludePods -> {"Input", "Position:SatelliteData", "SkyMap:SatelliteData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[15]=

Input interpretation:

International Space Station position

Results: Show DMS Show 3D
Orthographic projection



16.21° North 29.98° West (ocean)

(computed from orbital elements determined 203.7 minutes ago)

Current sky position from Olomouc, Olomoucky: Show DMS

(not currently visible)

altitude	-21.36° (below horizon)
azimuth	246° (WSW) (magnetic: 241.9°)
next rise	10:05 pm CET Tuesday, January 27, 2015
next set	10:10 pm CET Tuesday, January 27, 2015
constellation	Centaurus

2.2 Jaderné elektrárny

Po nehodě v jaderné elektrárně Fukushima v Japonsku 11. 3. 2011 se zvýšila pozornost o jaderné elektrárny ve světě. WolframAlpha má k dispozici databáze se všemi jadernými elektrárnami, včetně těch, které již nejsou v provozu, nebo těch, které jsou teprve ve výstavě. Při zadání příkazu "all nuclear reactors" získáme seznam všech jaderných reaktorů, základní informace o nich či jejich umístění na mapě světa.


```
In[16]:= WolframAlpha["all nuclear reactors",
  IncludePods -> {"Result", "Basic:NuclearReactorData", "WolframMap:NuclearReactorData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

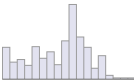
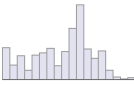
Out[16]=

Reactors:

More

Columbia | Bonus | Cooper | Fitzpatrick | Hallam | KANUPP | Lacrosse |
 Latina | Monticello | Phenix | Saxton | Trojan | Kewaunee | Palisades |
 Piqua | Borssele | Angra- 1 | Angra- 2 | Angra- 3 | Armenia- 1 | ... (total: 639)

Basic information:

net capacity	total	481 263 MW (megawatts) (electric power)
	median	878 MW (megawatts) (electric power)
	highest	1700 MW (megawatts) (electric power) (Taishan- 1 and Taishan- 2)
	lowest	0 MW (megawatts) (electric power) (Piqua)
	distribution	
gross capacity	total	510 100 MW (megawatts) (electric power)
	median	917 MW (megawatts) (electric power)
	highest	1750 MW (megawatts) (electric power) (Taishan- 1 and Taishan- 2)
	lowest	3 MW (megawatts) (electric power) (Saxton)
	distribution	

Local map:

31000 kilometers across | ▾

Non-metric



(based on current OpenStreetMap data)

```
In[17]:= WolframAlpha["all nuclear reactors", IncludePods -> {"Power:NuclearReactorData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[17]=

Power generation:

energy production	total	2.61×10^6 GW h/yr (gigawatt hours per year)
	median	5373 GW h/yr (gigawatt hours per year)
	highest	11 656 GW h/yr (gigawatt hours per year) (Isar- 2)
	lowest	0 GW h/yr (gigawatt hours per year) (12 nuclear power reactors)
	distribution	
energy availability factor	median	84.8%
	highest	100% (19 nuclear power reactors)
	lowest	-4.17% (JPDR)
	distribution	
load factor	median	82.9%
	highest	110.5% (Fugen ATR)
	lowest	0% (3 nuclear power reactors)
	distribution	
time on line	total	3.289×10^6 h/yr (hours per year)
	median	7506 h/yr (hours per year)
	highest	8780 h/yr (hours per year) (Bohunice- 2)
	lowest	0 h/yr (hours per year) (31 nuclear power reactors)
	distribution	
operational factor	median	87%
	highest	100.5% (Barsebäck- 2)
	lowest	0% (3 nuclear power reactors)
	distribution	

Kliknutím na jeden z reaktorů v tabulce (např. Kewaunee) získáme veškeré dostupné informace o tomto reaktoru, např. vlastníka reaktoru, výkon reaktoru, seznam okolních měst nebo datum, kdy byl zahájen provoz vybraného reaktoru.

```
In[18]:= WolframAlpha["Kewaunee nuclear reactor",
  IncludePods -> {"Basic:NuclearReactorData", "Power:NuclearReactorData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[18]=

Basic information: [More](#)

nuclear reactor site	Kewaunee (nuclear power station)
location	Kewaunee County
reactor type	PWR (Pressurized Light- Water- Moderated and Cooled Reactor)
owner	Dominion Generation
status	operational
net capacity	566 MW (megawatts) (electric power)
first commercial operation	16. 6. 1974 (40 years ago)

[+ Definitions](#)

Power generation: [Show history](#)

energy production	4595 GW h/yr (gigawatt hours per year) (lifetime total: 143 834 GW h)
energy availability factor	92.2% (lifetime average: 83.9%)
load factor	93.1% (lifetime average: 83.7%)
time on line	8068 h/yr (hours per year) (lifetime total: 280 243 h)
operational factor	92.1% (lifetime average: 84.8%)

(2011)

[+ Units](#)
[+ Definitions](#)

Kromě toho, že můžeme získat informace o konkrétním reaktoru, často existuje několik reaktorů postavených na jednom místě, o kterých můžeme získat souhrnné informace, např. příkazem “*Dresden nuclear power station*” získáme informace o všech třech reaktorech Dresden-1, Dresden-2 a Dresden-3.

```
In[19]:= WolframAlpha["Dresden nuclear power station",
  IncludePods -> {"Basic:NuclearReactorData", "Power:NuclearReactorData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Basic information:

[More](#)

	Dresden- 1	Dresden- 2	Dresden- 3
nuclear reactor site	Dresden (nuclear power station)	Dresden (nuclear power station)	Dresden (nuclear power station)
location	Grundy County	Grundy County	Grundy County
reactor type	BWR (Boiling Light- Water-Cooled and Moderated Reactor)	BWR (Boiling Light- Water-Cooled and Moderated Reactor)	BWR (Boiling Light- Water-Cooled and Moderated Reactor)
owner		Exelon	Exelon
status	permanent shutdown	operational	operational
net capacity	197 MW (megawatts) (electric power)	867 MW (megawatts) (electric power)	867 MW (megawatts) (electric power)
first commercial operation	4. 7. 1960 (54 years ago)	9. 6. 1970 (44 years ago)	16. 11. 1971 (43 years ago)
shutdown date	31. 10. 1978 (36 years ago)		

[+ Definitions](#)

Out[19]=

Power generation:

[Show history](#)

	Dresden- 1	Dresden- 2	Dresden- 3
energy production	762 GW h/yr (gigawatt hours per year) (lifetime total: 7223 GW h) (1978)	7181 GW h/yr (gigawatt hours per year) (lifetime total: 201 649 GW h) (2011)	7533 GW h/yr (gigawatt hours per year) (lifetime total: 194 752 GW h) (2011)
energy availability factor	53.1% (lifetime average: 67.3%) (1978)	93% (lifetime average: 78.6%) (2011)	100% (lifetime average: 75.4%) (2011)
load factor	53% (lifetime average: 44.66%) (1978)	94.6% (lifetime average: 69.2%) (2011)	99.2% (lifetime average: 69.2%) (2011)
time on line	6357 h/yr (hours per year) (lifetime total: 54 653 h) (1978)	8150 h/yr (hours per year) (lifetime total: 280 663 h) (2011)	8760 h/yr (hours per year) (lifetime total: 270 809 h) (2011)
operational factor	87.1% (lifetime average:) (1978)	93% (lifetime average: 76.8%) (2011)	100% (lifetime average: 76.5%) (2011)

[+ Units](#)

[+ Definitions](#)

Pokud chceme znát vývoj jaderné energetiky, můžeme ve WolframAlpha zadat příkaz “*nuclear reactors under construction*”, kde vidíme seznam jaderných reaktorů ve výstavbě.

```
In[20]:= WolframAlpha["nuclear reactors under construction",
  IncludePods -> {"Result", "WolframMap:NuclearReactorData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Reactors:

[More](#)

Angra- 3 | Atucha- 2 | Changjiang- 1 | Changjiang- 2 | Fangchenggang- 1 |
 Fangjiashan- 1 | Fangjiashan- 2 | Fuqing- 1 | Fuqing- 2 | Fuqing- 3 |
 Haiyang- 1 | Haiyang- 2 | Hongyanhe- 1 | Hongyanhe- 2 | Hongyanhe- 3 |
 Hongyanhe- 4 | Ningde- 1 | Ningde- 2 | Ningde- 3 | Ningde- 4 | ... (total: 60)

Local map:

32000 kilometers across | ▾

Non-metric



(based on current OpenStreetMap data)

Out[20]=

3 WolframAlpha pro chemiky

3.1 Chemické prvky a chemické reakce

WolframAlpha obsahuje velké množství chemických dat a poskytuje rychlé výpočty, které ušetří uživatelům značné množství času. S WolframAlpha lze např. porovnávat a zjišťovat informace o chemických prvcích. Zadáním příkazu "carbon" můžeme zjistit základní údaje o uhlíku, jeho umístění v periodické tabulce, dále termodynamické, materiálové nebo elektromagnetické vlastnosti. Jednoduše lze také zobrazit různé chemické reakce, jejichž řešení lze také zobrazit ve formátu Step-by-step, kde můžeme pozorovat jednotlivé kroky řešení, ne pouze samotný výsledek.

```
In[21]:= WolframAlpha["carbon",
  IncludePods -> {"PeriodicTableLocation:ElementData", "Elemental2:ElementData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Periodic table location:

Larger

H 1																	He 2																														
Li 3	Be 4											B 5	C 6	N 7	O 8	F 9	Ne 10																														
Na 11	Mg 12											Al 13	Si 14	P 15	S 16	Cl 17	Ar 18																														
K 19	Ca 20	Sc 21	Ti 22	V 23	Cr 24	Mn 25	Fe 26	Co 27	Ni 28	Cu 29	Zn 30	Ga 31	Ge 32	As 33	Se 34	Br 35	Kr 36																														
Rb 37	Sr 38	Y 39	Zr 40	Nb 41	Mo 42	Tc 43	Ru 44	Rh 45	Pd 46	Ag 47	Cd 48	In 49	Sn 50	Sb 51	Te 52	I 53	Xe 54																														
Cs 55	Ba 56		Hf 72	Ta 73	W 74	Re 75	Os 76	Ir 77	Pt 78	Au 79	Hg 80	Tl 81	Pb 82	Bi 83	Po 84	At 85	Rn 86																														
Fr 87	Ra 88		Rf 104	Db 105	Sg 106	Bh 107	Hs 108	Mt 109	Ds 110	Rg 111	Cn 112	Uut 113	Fl 114	Uup 115	Lv 116	Uus 117	Uuo 118																														
<table border="1"> <tbody> <tr> <td>La 57</td> <td>Ce 58</td> <td>Pr 59</td> <td>Nd 60</td> <td>Pm 61</td> <td>Sm 62</td> <td>Eu 63</td> <td>Gd 64</td> <td>Tb 65</td> <td>Dy 66</td> <td>Ho 67</td> <td>Er 68</td> <td>Tm 69</td> <td>Yb 70</td> <td>Lu 71</td> </tr> <tr> <td>Ac 89</td> <td>Th 90</td> <td>Pa 91</td> <td>U 92</td> <td>Np 93</td> <td>Pu 94</td> <td>Am 95</td> <td>Cm 96</td> <td>Bk 97</td> <td>Cf 98</td> <td>Es 99</td> <td>Fm 100</td> <td>Md 101</td> <td>No 102</td> <td>Lr 103</td> </tr> </tbody> </table>																		La 57	Ce 58	Pr 59	Nd 60	Pm 61	Sm 62	Eu 63	Gd 64	Tb 65	Dy 66	Ho 67	Er 68	Tm 69	Yb 70	Lu 71	Ac 89	Th 90	Pa 91	U 92	Np 93	Pu 94	Am 95	Cm 96	Bk 97	Cf 98	Es 99	Fm 100	Md 101	No 102	Lr 103
La 57	Ce 58	Pr 59	Nd 60	Pm 61	Sm 62	Eu 63	Gd 64	Tb 65	Dy 66	Ho 67	Er 68	Tm 69	Yb 70	Lu 71																																	
Ac 89	Th 90	Pa 91	U 92	Np 93	Pu 94	Am 95	Cm 96	Bk 97	Cf 98	Es 99	Fm 100	Md 101	No 102	Lr 103																																	

Out[21]=

Basic elemental properties:

More

symbol	C
atomic number	6
atomic mass	12.0107 u (unified atomic mass units)

```
In[22]:= WolframAlpha["carbon", IncludePods ->
  {"Thermal:ElementData", "Material:ElementData", "Electromagnetic:ElementData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[22]=

Material properties: More

density	buckminsterfullerene	1.65 g/cm ³
	activated charcoal	2.26 g/cm ³
	diamond	3.5 g/cm ³
Mohs hardness	0.5 (between talc and gypsum)	
sound speed	18 350 m/s (rank: 1 st)	
thermal expansion	7.1 × 10 ⁻⁶ K ⁻¹ (rank: 49 th)	
thermal conductivity	140 W/(mK) (rank: 12 th)	

(properties at standard conditions) + Units

Electromagnetic properties: More

electrical type	conductor	
resistivity	1 × 10 ⁻⁵ Ωm (ohm meters)	
magnetic type	diamagnetic	
color	■ (black)	
refractive index	activated charcoal	2.417

```
In[23]:= WolframAlpha["CH4 + 2O2 -> CO2 + 2H2O",
  IncludePods -> {"Input", "NewBalancedEquation:ChemicalReactionData",
  "ReactionStructures:ChemicalReactionData", "SubstanceProperties:ChemicalData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[23]=

Input interpretation:

CH₄ (methane) + 2 O₂ (oxygen) → CO₂ (carbon dioxide) + 2 H₂O (water)

Structures: Skeletal structure | ▾

```
In[24]= WolframAlpha["ethanol + oxygen -> water + carbon dioxide",
  IncludePods -> {"Input", "BalancedEquationPod:ChemicalReactionData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"BalancedEquationPod:ChemicalReactionData__Step-by-step",
    "BalancedEquationPod:ChemicalReactionData__Show all steps"}]
```

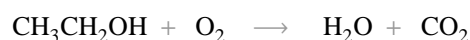
Input interpretation:



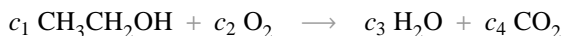
Balanced equation:

Start over

Balance the chemical equation:



Add coefficients to all the molecules:



The number of C, H, and O atoms on both sides of the reaction must be equal:

$$\text{C: } 2 c_1 = c_4$$

$$\text{H: } 6 c_1 = 2 c_3$$

$$\text{O: } c_1 + 2 c_2 = c_3 + 2 c_4$$

Since the coefficients are only determined up to a multiplicative constant, set $c_1 = 1$ and solve for the coefficients:

$$c_1 = 1$$

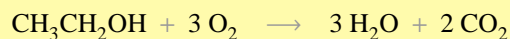
$$c_2 = 3$$

$$c_3 = 3$$

$$c_4 = 2$$

Since the coefficients are all integers with a greatest common denominator equal to 1, substitute the coefficients into the chemical reaction to obtain the balanced equation:

Answer:



Další silnou stránkou WolframAlpha je zobrazení chemických látek, jejich vlastností, strukturní vzorec a také 3D interaktivní model.

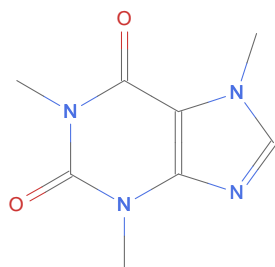
```
In[25]= WolframAlpha["caffeine", IncludePods ->
  {"ChemicalNamesFormulas:ChemicalData", "StructureDiagramPod:ChemicalData",
    "3DStructure:ChemicalData", "Basic:ChemicalData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```


Chemical names and formulas:

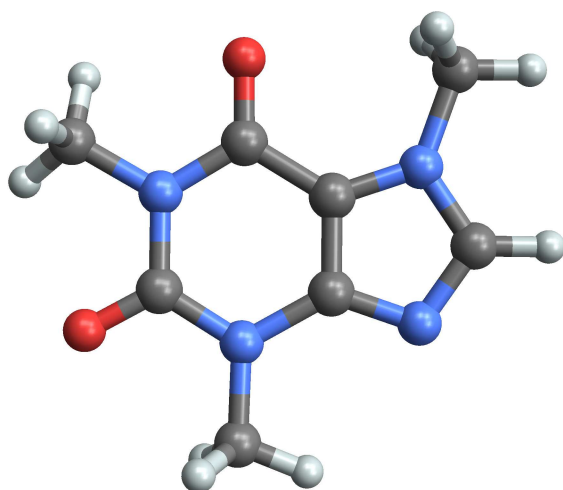
[More](#)

formula	$C_8H_{10}N_4O_2$
name	caffeine
IUPAC name	1,3,7-trimethylpurine-2,6-dione

Structure diagram:

[Skeletal structure](#) | ▾[Show bond information](#)[Step-by-step](#)

3D structure:

[Show space filling](#)

Basic properties:

molar mass	194.191 g/mol
phase	solid (at STP)
melting point	235.3 °C
density	1.23 g/cm ³

[Units](#)

Out[25]=

4 WolframAlpha pro geografii

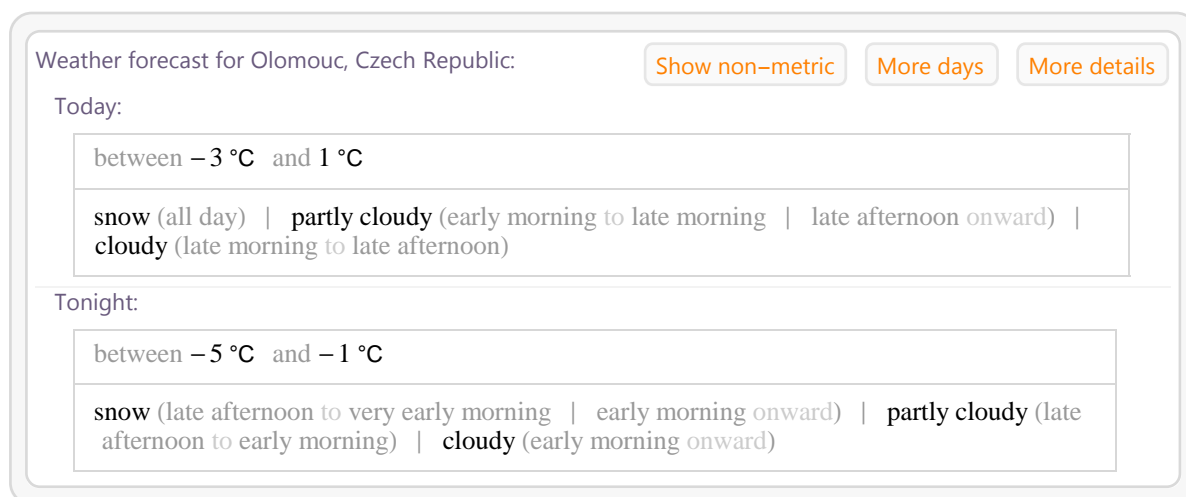
4.1 Počasí

Počasí a jeho předpověď hraje v našich běžných životech důležitou roli. Např. při cestě do práce nebo plánování letní či zimní dovolené. WolframAlpha nabízí řadu užitečných nástrojů, které zodpoví naše otázky o počasí. Můžeme získat aktuální hodnoty, ale i historická data z meteorologických stanic umístěných po celém světě.

Např. stačí zadat příkaz *"weather"* a WolframAlpha na základě vaší IP adresy vypíše aktuální teplotu, relativní vlhkost, rychlost větru, dále se můžeme dovědět předpověď na aktuální den a následující noc, ale také porovnat teplotu v den dotazu s teplotami ve stejné dny, ale několik desítek let zpátky. Z těchto historických dat, se zobrazí maximální a minimální hodnota, nebo průměr všech hodnot.

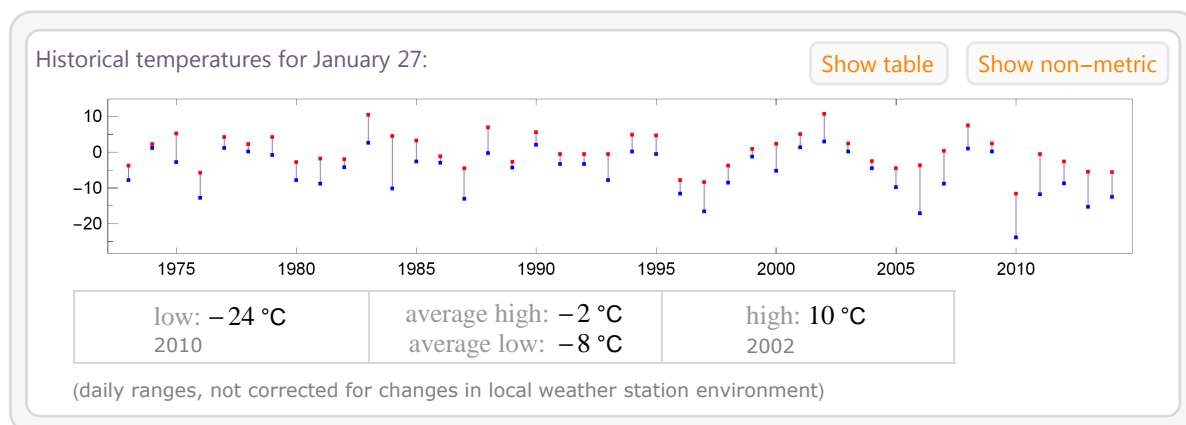
```
In[26]= WolframAlpha["weather", IncludePods -> "WeatherForecast:WeatherData",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[26]=



```
In[27]= WolframAlpha["weather", IncludePods -> "HistoricalTemperature:WeatherData",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[27]=



WolframAlpha umožňuje velmi jednoduše porovnávat data z různých míst. Pokud plánujete např. dovolenou není obtížné získat aktuální teplotní hodnoty z různých míst světa, nebo srovnat vývoj teploty v těchto místech za posledních několik let. Např. při zadání příkazu *"Weather Florence, Italy, Miami, Florida, Prague"* získáme informace o aktuálním počasí ve Florencii, Miami a Praze a lze se snadno podívat na vývoj teploty v těchto městech v např. posledních 10 letech nebo za poslední měsíc.

```
In[28]:= WolframAlpha["Weather Florence, Italy, Miami, Florida, Prague",
  IncludePods -> "WeatherForecast:WeatherData", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[28]=

Weather forecast: [Show non-metric](#) [More days](#) [More details](#)

Today:

Florence, Italy	between 0 °C and 6 °C clear (all day) snow (late afternoon onward)
Miami, United States	between 14 °C and 21 °C clear (all day)
Prague, Czech Republic	0 °C snow (all day) cloudy (early morning to late morning) partly cloudy (late morning to late afternoon) few clouds (late afternoon onward)

Tonight:

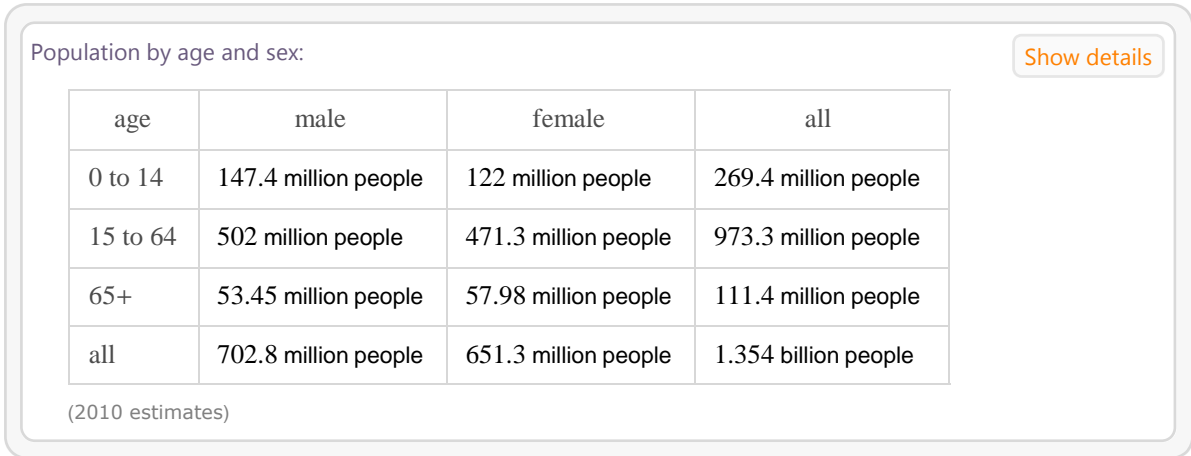
Florence, Italy	between -1 °C and 3 °C snow (late afternoon to early morning) clear (all night)
Prague, Czech Republic	between -2 °C and 1 °C snow (all night) few clouds (late afternoon to evening early morning) partly cloudy (evening to night very early morning to early morning) cloudy (late night to very early morning early morning onward)

4.2 Statistiky obyvatelstva

WolframAlpha umožňuje zobrazit podrobné statistiky populace v určité zemi. Např. při zadání “*China population distribution*” získáme rozložení obyvatelstva Číny podle věku a pohlaví.

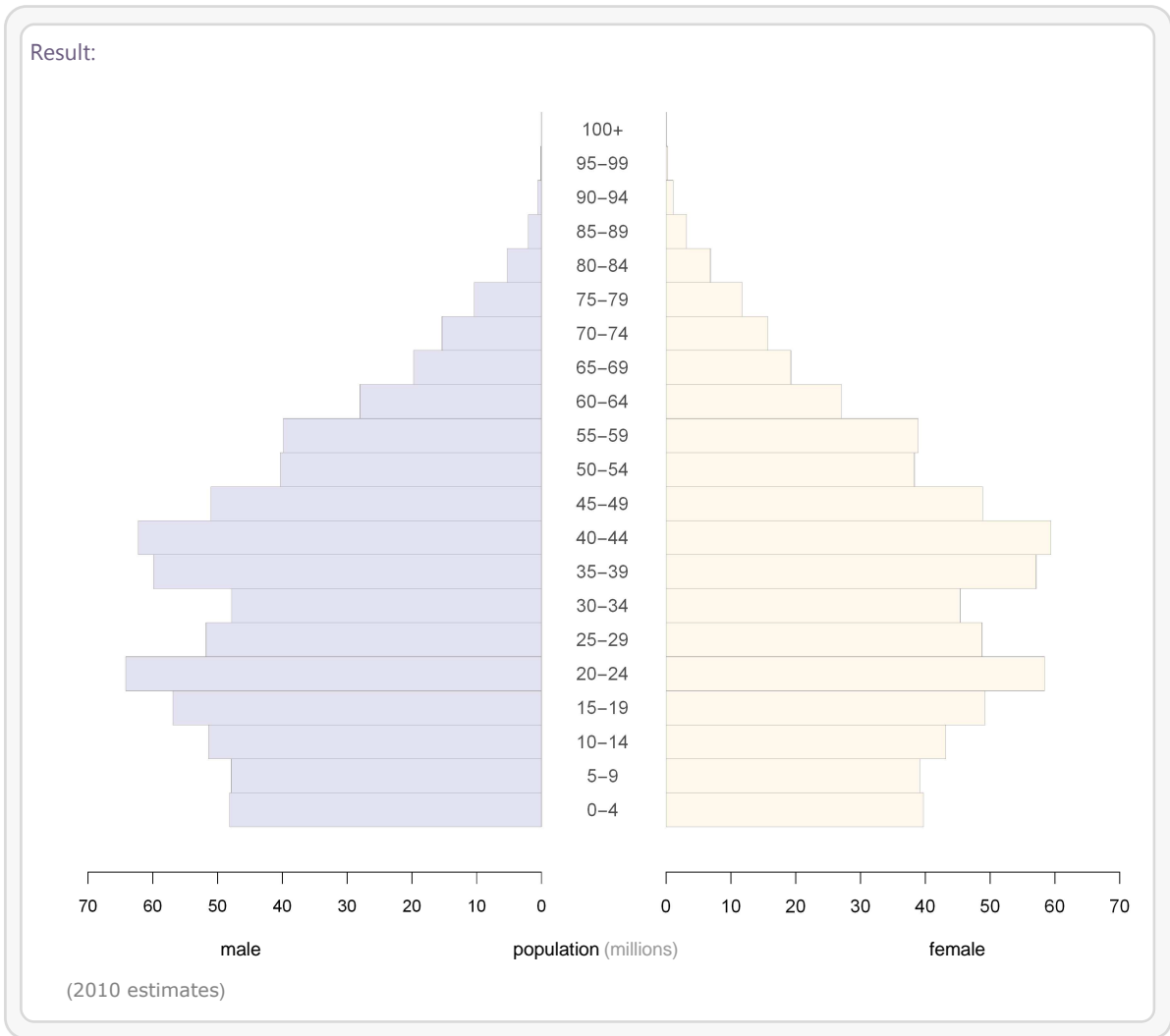
```
In[29]:= WolframAlpha["China population distribution",
  IncludePods -> {"AgeDistributionGrid:AgeDistributionData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[29]=



```
In[30]:= WolframAlpha["China population distribution",
  IncludePods -> {"AgeDistributionPyramidGraphic:AgeDistributionData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[30]=



Ve WolframAlpha můžeme také porovnat údaje z více států dohromady a např. prozkoumat, jak rychle zvolené země budou

stárnout. Např. zadáním příkazu "US, China, India population fraction age 65 in 2030" získáme odhad počtu obyvatel starších 65 let, kteří budou v roce 2030 žít v USA, Číně a Indii.

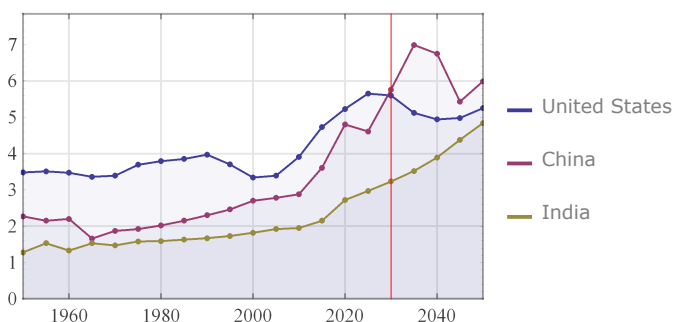
```
In[31]:= WolframAlpha["US, China, India population fraction age 65 in 2030",
  IncludePods -> {"Result", "PopulationSubsetFractionHistory:AgeDistributionData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Results:

United States	5.6%
China	5.76%
India	3.23%

(2030 estimates)

Population fractions history:



(from 1950 to 2050) (in percent)

Další zajímavou informací, kterou WolframAlpha zvládne velmi rychle vyřešit, je vypočítat např. vzájemný poměr žáků středních škol a učitelů a tyto výsledky zobrazit pro dvě různé země, např. Japonsko a Jižní Koreu.

```
In[32]:= WolframAlpha["High school student/teacher ratio in Japan vs. South Korea",
  IncludePods -> {"Input", "Result"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

Japan	student/teacher ratio	secondary education
South Korea		

[Definitions](#)

Results:

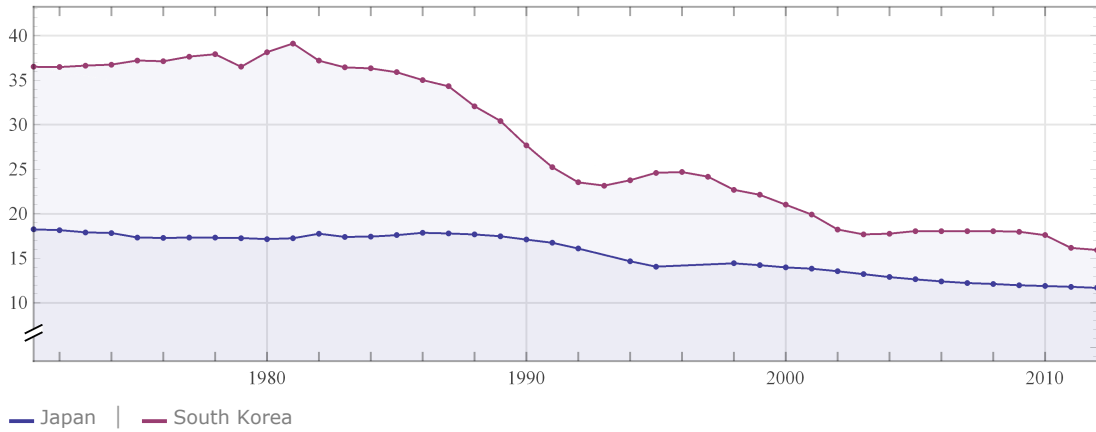
Japan	11.68
South Korea	15.94

(2012 estimates)

```
In[33]:= WolframAlpha["High school student/teacher ratio in Japan vs. South Korea",
  IncludePods -> {"History:PupilTeacherRatio:WorldDevelopmentData",
    "PupilTeacherRatio:WorldDevelopmentData", "Pupils:WorldDevelopmentData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Student/teacher ratio history:

[Log scale](#)



(from 1971 to 2012)

Student/teacher ratio:

	Japan	South Korea
primary school	17.09 (world rank: 78 th)	17.87 (world rank: 89 th)
secondary school	11.68 (world rank: 55 th)	15.94 (world rank: 108 th)

(2012 estimate)

[Definitions](#)

Students:

	primary school	secondary school	college
Japan	6.924 million people (39% of all students) (world rank: 19 th) (2012)	7.288 million people (40% of all students) (world rank: 14 th) (2012)	3.881 million people (21% of all students) (world rank: 8 th) (2011)
South Korea	2.959 million people (30% of all students) (world rank: 46 th) (2012)	3.783 million people (37% of all students) (world rank: 28 th) (2012)	3.356 million people (32% of all students) (world rank: 10 th) (2011)
	non-vocational secondary school		vocational secondary school
Japan	6.44 million people (world rank: 12 th) (2012)		848 498 people (world rank: 12 th) (2012)
South Korea	3.408 million people (world rank: 27 th) (2012)		374 676 people (world rank: 27 th) (2012)

[Definitions](#)

Out[33]=

4.3 Mapy

V geografii jsou velmi důležité aktuální mapy, které lze jednoduše zobrazit a srovnávat. WolframAlpha obsahuje velké množství map. Např. příkazem *“Argentina map”* zobrazíme mapu Argentiny, příkazem *“population”* lze zobrazit mapu s rozložením populace jednotlivých států. Stejně tak můžeme zobrazit polohu některé významné budovy a stavby na mapě světa, např. zadáním příkazu *“Eiffel Tower, Burj Khalifa”, Willis Tower*.

```
In[34]:= WolframAlpha["Argentina map", IncludePods -> "Location:CountryData",  
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Result:

Show mesh



Out[34]=



Satellite image »

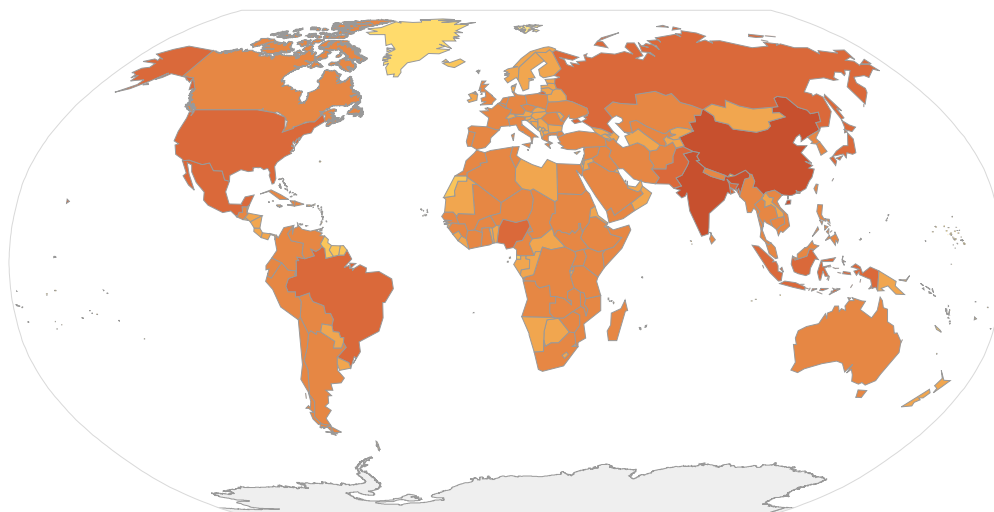
```
In[35]= WolframAlpha["population",
  IncludePods -> {"Result", "PropertyMap:Population:CountryData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Summary:

total	7.22 billion people
highest	1.36 billion people (China)
lowest	54 people (Pitcairn Islands)

(2011, 2012, 2013, and 2014 estimates)

Population map:



10 to 100
 10000 to 100000
 10 million to 100 million
 100 to 1000
 100000 to 1 million
 100 million to 1 billion
 1000 to 10000
 1 million to 10 million
 1 billion to 10 billion
 (in people)

```
In[36]= WolframAlpha["Eiffel Tower, Burj Khalifa, Willis Tower", IncludePods ->
  {"Basic2:BuildingData", "DesignAndConstruction:BuildingData", "Image:BuildingData"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```


Basic information:

Show non-metric




	Eiffel Tower	Burj Khalifa	Willis Tower
cities	Paris, Ile- de- France, France	Dubai, United Arab Emirates	Chicago, Illinois, United States
height	324 meters (city rank: 1 st national rank: 1 st world rank: 52 nd)	828 meters (city rank: 1 st national rank: 1 st world rank: 1 st)	442 meters (city rank: 1 st national rank: 2 nd world rank: 15 th)
floors	3	163	108
floor area		334 000 m ² (square meters)	423 658 m ² (square meters)
construction cost		\$4 billion (US dollars)	
completion date	1889 (126 years ago)	January 2010 (5 years ago)	1974 (41 years ago)

Design:

	Eiffel Tower	Burj Khalifa	Willis Tower (Sears Tower)
architect	Stephen Sauvestre	Adrian Smith George J. Efstathiou Marshall Strabala William F. Baker	Bruce Graham Fazlur Khan
firm	Barbier, Benard and Turenne Eiffel & Cie	Skidmore, Owings and Merrill	Skidmore, Owings and Merrill
style		Islamic architecture High-tech architecture Futurist architecture Expressionist architecture	International Style

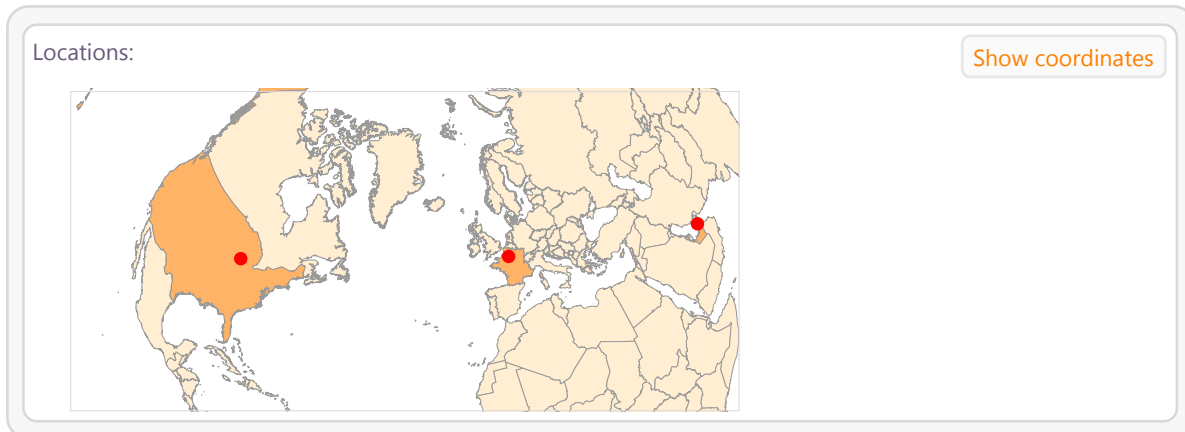
Out[36]=

Images:

Eiffel Tower	Burj Khalifa	Willis Tower (Sears Tower)
		

```
In[37]:= WolframAlpha["Eiffel Tower, Burj Khalifa, Willis Tower",
  IncludePods -> "Location:BuildingData", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[37]=



4.4 Tornáda, sopečné erupce a zemětřesení

Tornádo je silně rotující větrný vír. Tornáda se vyskytují po celém světě, ale jejich největší výskyt je v USA. WolframAlpha umožňuje zobrazit informace o tornádech (především v USA) od roku 1950 do současnosti. Pro popis tornád se používá Fujitova stupnice, která má šest stupňů podle síly větru, od nejslabšího typu F0 až po nejsilnější typ F5.

```
In[38]:= WolframAlpha["tornadoes within 25 km of St.Louis",
  IncludePods -> {"Input", {"Result", 1}, {"Result", 2}, {"Result", 3},
  "DisasterTemplateSummaryReport"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

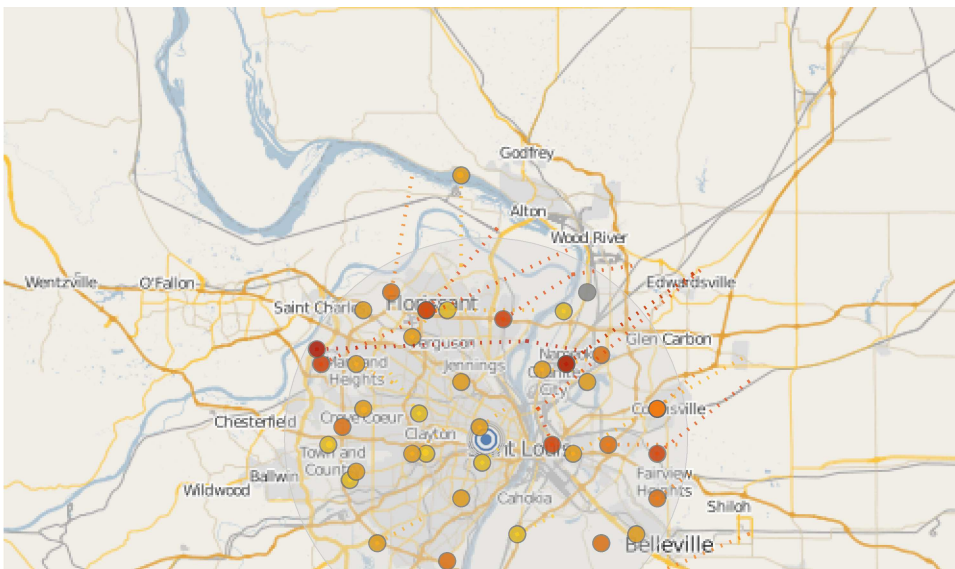
Input interpretation:

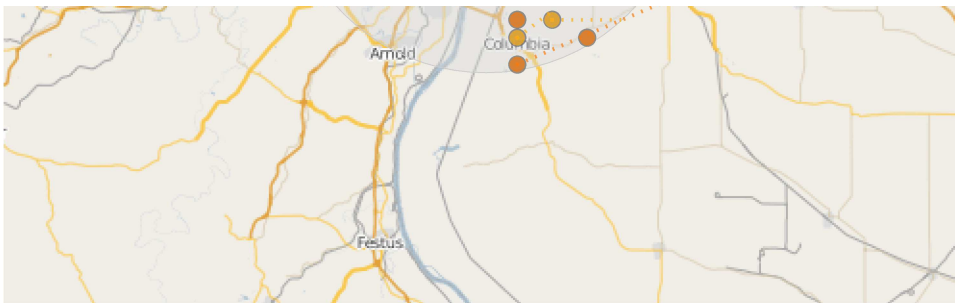
tornadoes	within 25 km (kilometers)
	near Saint Louis, Missouri, United States

Results:

Local map:

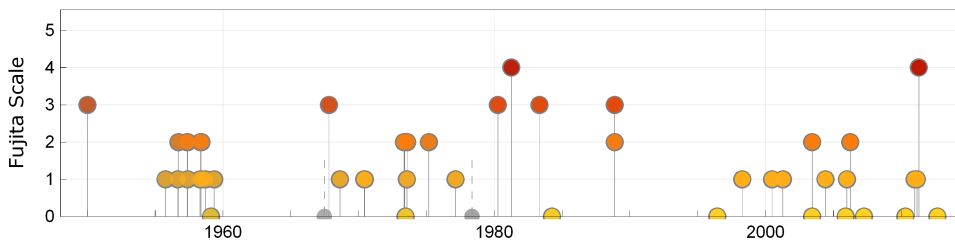
120 kilometers across | Non-metric





(based on current OpenStreetMap data)

Timeline:



Out[38]=

List:

|

Fujita scale	date	touchdown location
● F4	Fri, Apr 22, 2011 6:59 pm CST (3.8 years ago)	24 km WNW from Saint Louis, Missouri
● F4	Fri, Apr 3, 1981 10:50 pm CST (33.8 years ago)	14 km NE from Saint Louis, Missouri
● F3	Sun, May 1, 1983 7:20 pm CST (31.8 years ago)	8 km E from Saint Louis, Missouri

Summary:

injuries	total	87 people
	highest	32 people (April 3, 1981 10:50 pm CST (1 km SSW from Granite City, Illinois))
	lowest	0 people (35 tornadoes)
fatalities	total	2 people
	highest	1 person (April 2, 2006 4:15 pm CST (2 km NNW from Swansea, Illinois) and May 30, 2004 4:20 pm CST (1 km WNW from Berkeley, Missouri))
	lowest	0 people (49 tornadoes)
losses	total	\$(50.45 million to 158.9 million) (US dollars)
	highest	\$(5 million to 50 million) (US dollars)

	(May 1, 1983 7:20 pm CST (2 km NW from East Saint Louis, Illinois) and April 3, 1981 10:50 pm CST (1 km SSW from Granite City, Illinois))
lowest	\$(50 to 500) (US dollars) (July 23, 1973 4:30 pm CST (0.4 km SSE from Northwoods, Missouri) and February 10, 1959 1:40 am CST (2 km NNE from Calverton Park, Missouri))
mean	\$(1.363 million to 4.294 million) (US dollars)

+ Units

```
In[39]= WolframAlpha["F4 tornadoes in Illinois",
  IncludePods -> {"Input", {"Result", 1}, {"Result", 2}, {"Result", 3},
  "DisasterTemplateSummaryReport"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

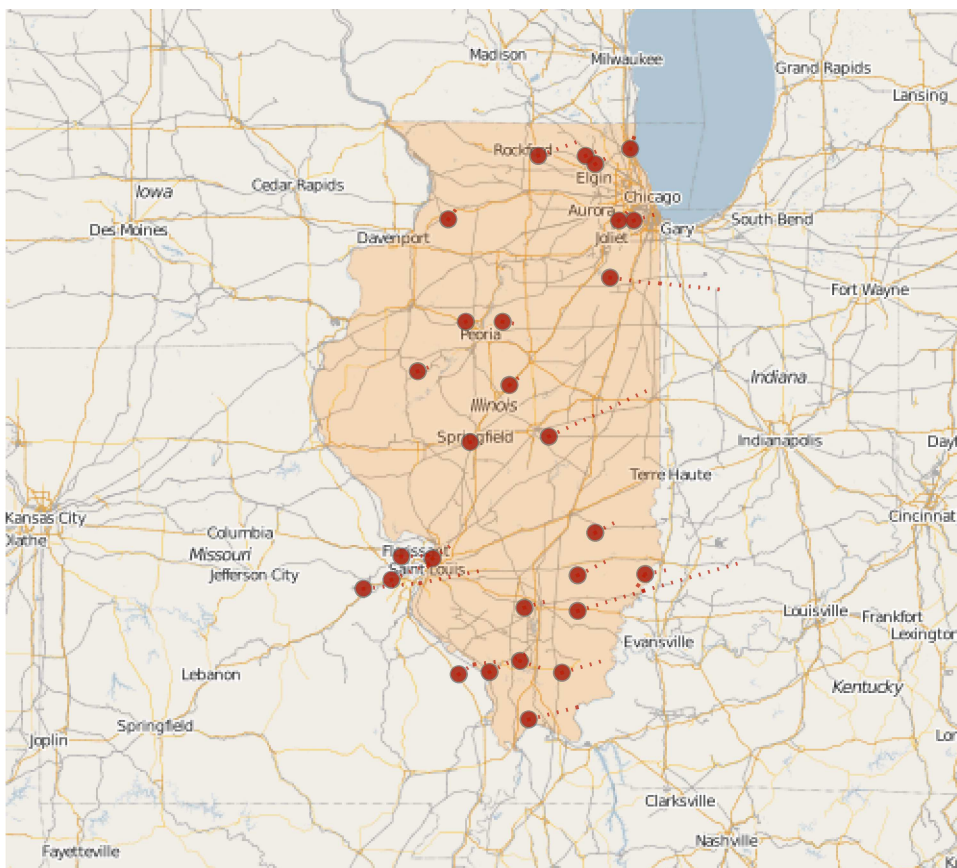
Input interpretation:

tornadoes	F4
	in Illinois (US state)

Results:

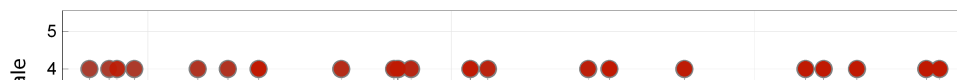
Local map:

950 kilometers across | Non-metric

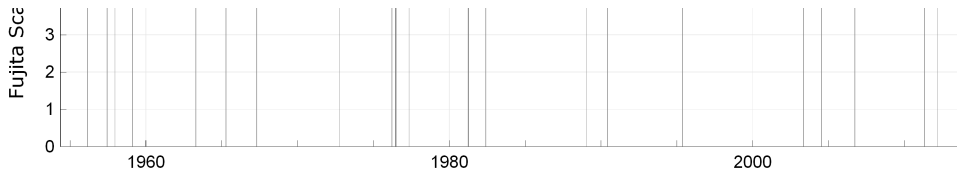


(based on current OpenStreetMap data)

Timeline:



Out[39]=



List:

Sort by intensity | ▾

More

Show coordinates

Show non-metric

Fujita scale	date	touchdown location
● F4	Fri, Apr 22, 2011 6:59 pm CST (3.8 years ago)	3 km NNW from Maryland Heights, Missouri
● F4	Fri, Sep 22, 2006 3:13 pm CST (8.4 years ago)	5 km WNW from Longtown, Missouri
● F4	Tue, Feb 10, 1959 1:40 am CST (56 years ago)	5 km NE from Eureka, Missouri

Summary:

injuries	total	2422 people
	highest	500 people (April 21, 1967 5:24 pm CST (0.5 km WSW from Palos Park, Illinois))
	lowest	0 people (March 29, 1981 6:30 pm CST (2 km NW from Prophetstown, Illinois) and June 29, 1976 1:45 pm CST (9 km SW from Dunlap, Illinois))
fatalities	total	129 people
	highest	33 people (April 21, 1967 5:24 pm CST (0.5 km WSW from Palos Park, Illinois))
	lowest	0 people (13 tornadoes)
losses	total	\$(137 million to 867.8 million) (US dollars)
	highest	\$(50 million to 500 million) (US dollars) (May 29, 1982 2:05 pm CST (2 km NNW from Cambria, Illinois))
	lowest	\$(50 000 to 500 000) (US dollars) (4 tornadoes)
	mean	\$(5.269 million to 33.38 million) (US dollars)

Units

```
In[40]:= WolframAlpha["F5 tornadoes after 1950",
  IncludePods -> {"Input", {"Result", 1}, {"Result", 2}, {"Result", 3},
  "DisasterTemplateSummaryReport"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

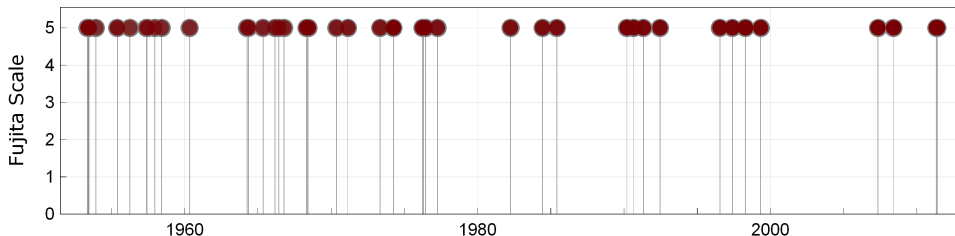
tornadoes	after 1950
	F5

Results:

Local map:



Timeline:



List:

[Sort by intensity](#) | [More](#)
[Show coordinates](#) | [Show non-metric](#)

Out[40]=

Fujita scale	date	touchdown location
● F5	Thu, Jul 18, 1996 6:05 pm CST (18.5 years ago)	6 km WNW from Oakfield, Wisconsin
● F5	Thu, Jun 7, 1984 11:41 pm CST (30.7 years ago)	3 km SSE from Ridgeway, Wisconsin
● F5	Wed, Jun 4, 1958 5:30 pm CST (56.7 years ago)	2 km W from Eau Galle, Wisconsin

Summary:

injuries	total	12 744 people
	highest	1150 people (April 3, 1974 1:30 pm CST (3 km ESE from Bellbrook, Ohio) and May 22, 2011 4:34 pm CST (5 km SE from Galena, Kansas))
	lowest	0 people (3 tornadoes)
fatalities	total	1323 people
	highest	158 people (May 22, 2011 4:34 pm CST (5 km SE from Galena, Kansas))
	lowest	0 people (6 tornadoes)
losses	total	\$(6.099 billion to 9.612 billion) (US dollars)
	highest	\$2.8 billion (US dollars) (May 22, 2011 4:34 pm CST (5 km SE from Galena, Kansas))
	lowest	\$(50 000 to 500 000) (US dollars) (7 tornadoes)
	mean	\$(117.3 million to 184.8 million) (US dollars)

Units

```
In[41]= WolframAlpha["volcanic eruptions in 40N, 14E within 200 km",
  IncludePods -> {"Input", {"Result", 1}, {"Result", 2}, {"Result", 3},
  "DisasterTemplateSummaryReport"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

volcanic eruptions

within 200 km (kilometers)

near 40° North 14° East

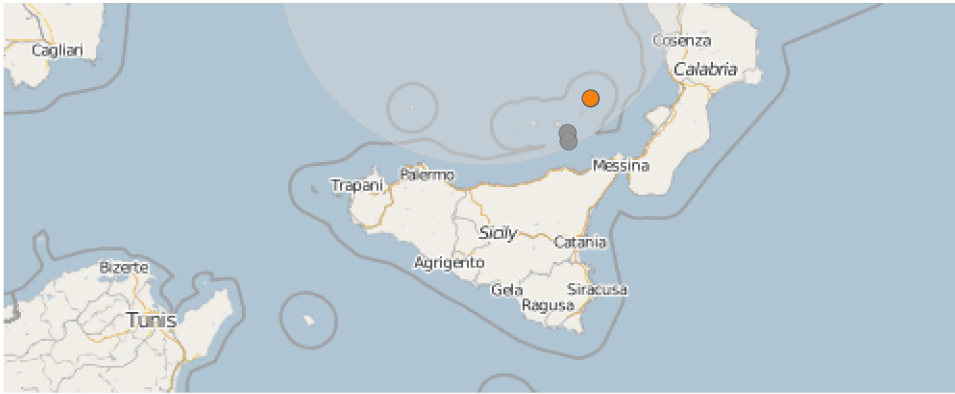
Results:

Local map:

940 kilometers across

Non-metric

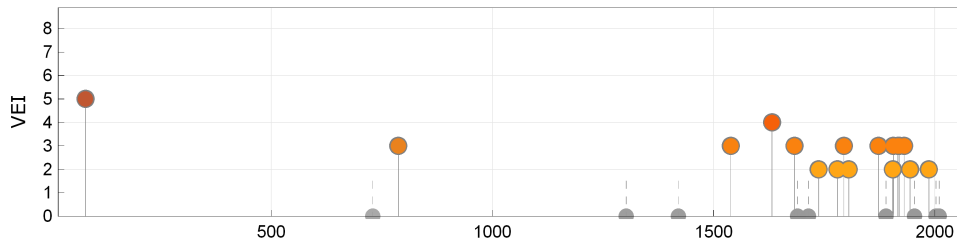




(based on current OpenStreetMap data)

Out[41]=

Timeline:



List:

Sort by date ▾

More

Show coordinates

Show non-metric

volcanic explosivity index	date	location
● (unavailable)	2010 (5.1 years ago)	Stromboli 43 km NE from Lipari, Sicily, Italy
● (unavailable)	Mon, Dec 30, 2002 (12.1 years ago)	Stromboli 43 km NE from Lipari, Sicily, Italy
● 2 (explosive)	Thu, Jul 24, 1986 (28.5 years ago)	Stromboli 43 km NE from Lipari, Sicily, Italy

Summary:

deaths	total	8330 people
	highest	4000 people (Vesuvius (1631))
	lowest	1 person (Stromboli (1986) and Vesuvius (1905))
injuries	total	312 people
	highest	300 people (Vesuvius (1906))
	lowest	1 person (Vesuvius (1905))

```
In[42]= WolframAlpha["volcanic eruptions near kamchatka, russia after 1750",
IncludePods -> {"Input", {"Result", 1}, {"Result", 2}, {"Result", 3}},
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```


Input interpretation:

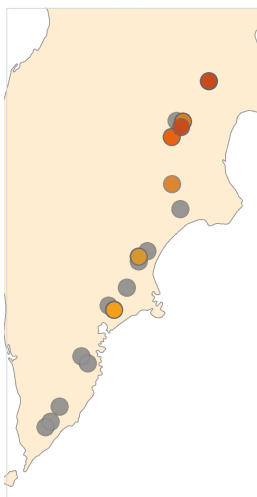
volcanic eruptions

after 1750

in Kamchatka, Russia (region)

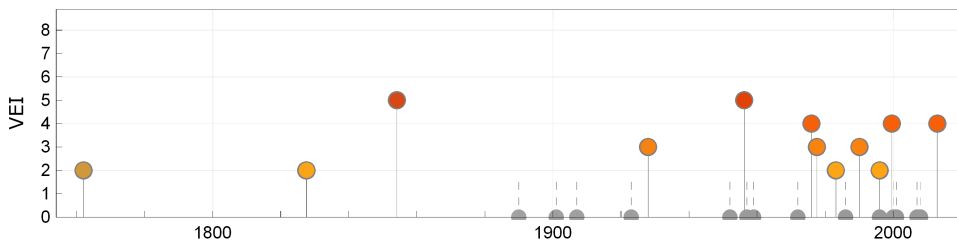
Results:

Local map:



Out[42]=

Timeline:



List:

Sort by date | ▾

More

Show coordinates

Show non-metric

volcanic explosivity index	date	location
● 4 (explosive)	Wed, Dec 12, 2012 (2.1 years ago)	Tolbachik 39 km SE from Kozyryovsk, Kamchatka, Russia
● (unavailable)	2008 (7.1 years ago)	Shiveluch 50 km NE from Klyuchi, Kamchatka, Russia
● (unavailable)	2007 (8.1 years ago)	Kliuchevskoi 30 km SSW from Klyuchi, Kamchatka, Russia

```
In[43]= WolframAlpha["most destructive volcanic eruption",
  IncludePods -> {"Result", "Properties:VolcanoEruptionData",
    "Fatalities:VolcanoEruptionData", "OrdinalRankings"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Result:

St. Helens (1980)
(\$2 billion (US dollars))

Eruption properties:

start date	27. 3. 1980
peak of activity	18. 5. 1980 (34 years ago)
end date	28. 10. 1986
dating technique	historical

Volcano properties:

name	St. Helens
location	United States
type	stratovolcano

Out[43]=

Effects:

cause of fatalities	pyroclastic flows, surges and direct blasts mudflows/lahars tephra
deaths	57 people
damage cost	\$2 billion (US dollars)

[+ Definitions](#)

Volcanic eruptions ranked by damage cost:

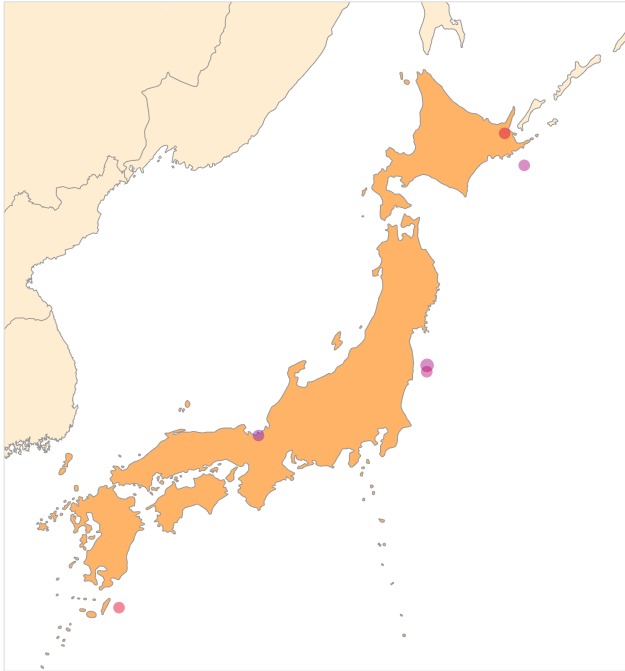
[More](#)

1	St. Helens (1980)	\$2 billion	
2	Merapi (2010)	\$600 million	
3	Rabaul (1994)	\$35 million	
4	Galunggung (1982)	\$15 million	
5	Etna (1981)	\$10 million	

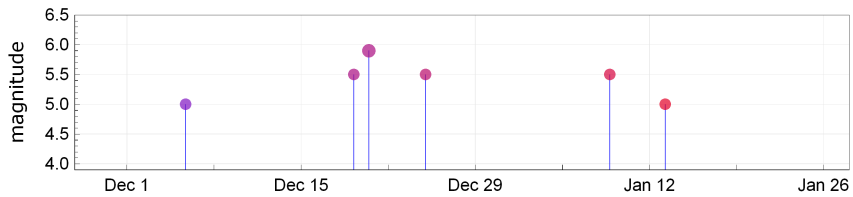
⋮

```
In[44]:= WolframAlpha["earthquakes in Japan", IncludePods -> "Results",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Results: Last 2 months Magnitude > 5 CET Show local map



Timeline:



List: Sort by magnitude More Show coordinates Show non-metric

magnitude	time	location
5.9	Sat, Dec 20, 2014 10:29 am CET (1.2 months ago)	49 km E of Okuma, Fukushima, Japan
5.5	Thu, Jan 8, 2015 07:42 pm CET (18.6 days ago)	14 km NW of Nakashibetsu, Hokkaido, Japan
5.5	Thu, Dec 25, 2014 12:06 am CET (1.1 months ago)	51 km ESE of Okuma, Fukushima, Japan

Out[44]=

5 WolframAlpha pro matematiky a ekonomy

5.1 Bitcoin

Bitcoin je virtuální měna, která není regulována bankou nebo státem. Bitcoin je produkován přes proces známý jako “dobývání”, a na rozdíl od mnoha jiných on-line transakcí je lze odesílat a přijímat anonymně z osoby na osobu. Bitcoinů mohou být uloženy v osobním počítači ve formě souboru s peněženkou nebo uchovávány pomocí služby třetí strany. Celkové množství Bitcoinů, které budou vytěženy, je asi 21 000 000. Růst se však postupně zpomaluje, a veškeré bitcoiny budou vytěženy v roce 2140 (drtivá většina však cca v roce 2030).

Pomocí WolframAlpha lze zobrazit informace o vývoji kurzu této měny, maximum a minimum za poslední rok. Je možné se podívat zpět v historii a zjistit hodnotu této měny např. 1. 1. 2011.

```
In[45]:= WolframAlpha["1 bitcoin",
  IncludePods -> {"LocalCurrencyConversion", "History", "CurrencyConversions"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Local currency conversion:

Kč6367.34 (Czech koruny) (at current quoted rate)

Exchange history for ₿1 (bitcoin):

Reverse rate

Log scale

Last year | ▾



Out[45]=

1-year minimum	Kč4482.49 (14. 1. 2015 13 days ago)
1-year maximum	Kč17 600 (28. 1. 2014 12 months ago)
1-year average	Kč9879.89 (annualized volatility: 78%)

Units

Additional currency conversions for ₿1 (bitcoin):

World currencies

Reverse rates

USD	\$259.51 (US dollars)
JPY	¥30 670 (Japanese yen)
EUR	€229.34 (euros)
CNY	¥ 1620.56 (Chinese yuan)

```
In[46]:= WolframAlpha["1 bitcoin on 1/1/11",
  IncludePods -> {"Input", "Result", "CurrencyConversions"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[46]=

Input interpretation:

฿1 (bitcoin) to	Saturday, January 1, 2011
-----------------	---------------------------

Result:

Kč5.61 (Czech koruny)

Currency conversions: World currencies

USD	22.85¢ (US cents)
EUR	20.19c (euro cents)
GBP	15.14p (British pence)
CNY	¥ 1.43 (Chinese yuan)
RUB	py615.46 (Russian rubles)
PLN	85.24 groszy (Polish groszy)

5.2 Využití WolframAlpha ve finanční matematice

System WolframAlpha umožňuje řešení celé řady aktuálních témat týkajících se různých typů úročení, poskytování hypotéčních, spotřebitelských či jiných úvěrů nebo práce s měnovými kurzy. Jedním ze základních témat finanční matematiky je problematika jednoduchého úročení. Budoucí hodnotu (FV - future value) kapitálu v tomto případě počítáme ze stále stejné počáteční částky (PV - present value). V následujícím příkladu vypočítáme budoucí hodnotu počáteční částky 950 Kč při roční úrokové míře 2% po dobu 5 let. Při změně vstupních hodnot je dále možné z budoucí hodnoty počítat hodnoty současné, či úrokovou míru, která za dané období zúročí počáteční kapitál na jeho budoucí hodnotu. V dalším příkladu pak uvádíme výpočet roční úrokové míry, která za 5 let zúročí částku 950 Kč na hodnotu 1125 Kč. Do WolframAlpha zapíšeme příkaz *“simple interest”*, poté zvolíme možnost *“future value”* a zapíšeme počáteční údaje, v druhém příkladě zvolíme místo *“future value”* příkaz *“interest rate”*.

```
In[47]= WolframAlpha["simple interest",
  IncludePods -> {"Input", "Result", "Equation"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*F.SimpleInterest.PV-_K%C4%8D950", "*F.SimpleInterest.i-_2+%25",
    "*F.SimpleInterest.n-_5"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input information:

simple interest	
present value	Kč950 (Czech koruny)
interest rate	2%
interest periods	5

Result:

future value	Kč1045.00 (Czech koruny)
--------------	--------------------------

Equation:

$FV = PV (1 + i n)$	
FV	future value
PV	present value
<i>i</i>	interest rate
<i>n</i>	interest periods

Out[47]=

```
In[48]= WolframAlpha["simple interest",
  IncludePods -> {"Input", "Result"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*FS-_*SimpleInterest.i--", "*F.SimpleInterest.FV-_K%C4%8D+1125",
    "*F.SimpleInterest.PV-_K%C4%8D+950", "*F.SimpleInterest.n-_5"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input information:

simple interest	
future value	Kč1125 (Czech koruny)
present value	Kč950 (Czech koruny)
interest periods	5

Result:

interest rate	3.684% = 0.03684
---------------	---------------------

Out[48]=

```
In[49]= WolframAlpha["simple interest",
  IncludePods -> {"Equation"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*FS-_*SimpleInterest.i--", "*F.SimpleInterest.FV-_K%C4%8D+1125",
    "*F.SimpleInterest.PV-_K%C4%8D+950", "*F.SimpleInterest.n-_5"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[49]=

Equation:

$FV = PV (1 + i n)$	
i	interest rate
FV	future value
PV	present value
n	interest periods

WolframAlpha dále umožňuje řešení problematiky složeného úročení, kde je počáteční kapitál úročen po dobu tvořenou více úrokovými obdobími. Úrok bude ke vkladu připsán vždy na konci roku a následující rok bude znovu spolu s vkladem úročen, vzniknou tedy úroky z úroků. V následujícím příkladu budeme počítat budoucí hodnotu částky \$800 při roční úrokové míře 6% po dobu 5 let se čtvrtletním připsáváním úroků (systém WolframAlpha umožňuje tuto hodnotu nastavit dle nejpoužívanějších typů připsávání úroků v praxi). Výsledek je dále automaticky převáděn na české koruny dle aktuálního měnového kurzu. Grafické výstupy umožňují porovnání závislosti úrokové míry na budoucí hodnotě, doby úročení na budoucí hodnotě, či závislost současné a budoucí hodnoty kapitálu. Do WolframAlpha zapíšeme příkaz *"compound interest"*, poté zvolíme možnost *"future value"* a zapíšeme počáteční údaje.

```
In[50]= WolframAlpha["compound interest",
  IncludePods -> {"Input", "Result"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*FVarOpt-_*PresentValueFutureValue.compoundingfreq--",
    "*FS-_*PresentValueFutureValue.FV--", "*F.PresentValueFutureValue.PV-_%24800",
    "*F.PresentValueFutureValue.i-_6+%25", "*F.PresentValueFutureValue.n-_5",
    "*FP.PresentValueFutureValue.compoundingfreq-_CompoundingFrequency%3AQuarterly"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[50]=

Input information:

present and future value	
present value	\$800 (US dollars)
interest rate	6%
interest periods	5
compounding frequency	quarterly

Result:

future value	\$1077.48 (US dollars) = Kč26430 (Czech koruny)
--------------	--

```
In[51]= WolframAlpha["compound interest", IncludePods ->
{"Equation", "FutureValueVs.InterestRate", "FutureValueVs.InterestPeriods"},
AppearanceElements -> {"Pods"}, InputAssumptions ->
{"*FVarOpt-*PresentValueFutureValue.compoundingfreq--",
"*FS-*PresentValueFutureValue.FV--", "*F.PresentValueFutureValue.PV-_%24800",
"*F.PresentValueFutureValue.i-_6+%25", "*F.PresentValueFutureValue.n-_5",
"*FP.PresentValueFutureValue.compoundingfreq-_CompoundingFrequency%3AQuarterly"},
TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

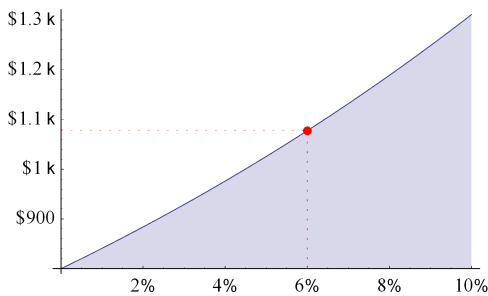
Equation:

$FV = PV \left(1 + \frac{i}{f}\right)^{fn}$	
FV	future value
PV	present value
<i>i</i>	interest rate
<i>n</i>	interest periods
<i>f</i>	compounding frequency

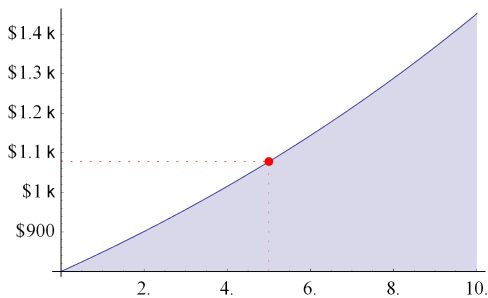
(assumes finite compounding)

Out[51]=

Future value vs. interest rate:

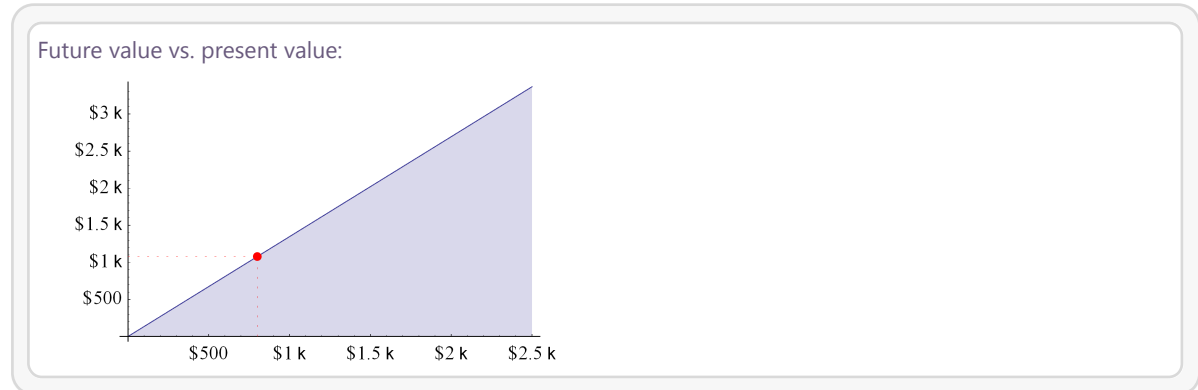


Future value vs. interest periods:




```
In[52]= WolframAlpha["compound interest",
  IncludePods -> {"FutureValueVs.PresentValue"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*FVarOpt-*PresentValueFutureValue.compoundingfreq--",
    "*FS-*PresentValueFutureValue.FV--", "*F.PresentValueFutureValue.PV-_%24800",
    "*F.PresentValueFutureValue.i-_6+%25", "*F.PresentValueFutureValue.n-_5",
    "*FP.PresentValueFutureValue.compoundingfreq-_CompoundingFrequency%3AQuarterly"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[52]=



V rámci problematiky týkající se složeného úročení se setkáváme s pojmem efektivní úroková míra. Jde o úrokovou míru, která poskytne za jedno roční úrokové období stejný úrok jako nominální úroková míra s častějším připisováním úroků. Systém WolframAlpha řeší výpočty efektivní úrokové míry pomocí vstupu "effective interest rate". Změnou vstupních hodnot je možné měnit cíl našich výpočtů (např. nominální úrokovou míru nebo různé frekvence úročení). Do WolframAlpha zapíšeme příkaz "effective interest rate", poté zapíšeme počáteční údaje.

```
In[53]= WolframAlpha["effective interest rate",
  IncludePods -> {"Equation", "InputValue", "Result"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[53]=

Equation:

$r_{\text{eff}} = \left(1 + \frac{r}{n}\right)^n - 1$	
r_{eff}	effective interest rate
r	nominal interest rate
n	compounding periods

(assumes finite compounding)

Input values:

nominal interest rate	8%
compounding periods	12

Result:

effective interest rate	8.3% = 0.083
-------------------------	-----------------

Jednou z dalších oblastí finanční matematiky je problematika splácení úvěrů. Dluh je možné splácet dle splátkového kalendáře splátkami stejné výše nebo konstantním úmorem (částka, která se skutečně odečte od posledního stavu dluhu). V prvním případě se s každou splátkou mění výše vypočteného úroku i úmoru. V druhém případě bude v každém období umořena stejná část dluhu.

Následující příklad řeší splácení úvěru ve výši 400000 Kč na dobu 5 let při roční úrokové míře 4 % splátkami stejné výše. Interval splácení je možné nastavit např. na měsíční frekvenci. Grafický výstup ukazuje jakou část celkové zaplacené částky zaujímá splacený úrok. Do WolframAlpha zapíšeme příkaz "loan"; poté zapíšeme počáteční údaje.

```
In[54]:= WolframAlpha["loan", IncludePods -> {"InputValue", "LoanPayments", "LoanTotals"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input values:

loan amount	Kč400 000 (Czech koruny)
loan period	5 years
annual percentage rate	4%
payment interval	monthly

Loan payments:

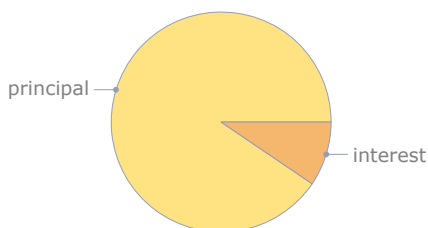
monthly payment	Kč7366.61
number of payments	60
time to first payment	1 month
effective interest rate	4.074%

(assuming the last payment is due the last day of the loan period)

+ Units

Loan totals:

principal paid	Kč400 000
total interest paid	Kč41 997
total amount paid	Kč441 997



+ Units

Nejčastější formou financování nemovitosti je hypoteční úvěr. Velikost poskytnuté půjčky je v současné době až sto procent. Úvěr se v současné době poskytuje na dobu 5 - 30 let a bývá obvykle splácen měsíčními anuitami. Automaticky tedy předpokládáme měsíční úročení. Následující příklad demonstruje průběh splácení úvěru ve výši 4000000 Kč po dobu 30 let při roční úrokové míře 2,85%. Výpočet pomocí WolframAlpha nabízí výši měsíční platby, efektivní úrokové míry a celkovou zaplacenou částku. Jedním z výstupů je rovněž umořovací plán hypotéky na 30 let. Do WolframAlpha zapíšeme příkaz "fixed rate mortgage"; poté zapíšeme počáteční údaje.

```
In[55]= WolframAlpha["fixed rate mortgage",
  IncludePods -> {"Input", "MonthlyPayments", "MortgageTotals", "PaymentsAndBalances"},
  AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*F.FixedRateMortgage.MA-_4000000+Czech+koruny",
    "*F.FixedRateMortgage.MP-_30+yr", "*F.FixedRateMortgage.APR-_2.85+%25"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input information:

fixed rate mortgage	
loan amount	Kč4 million (Czech koruny)
loan period	30 years
annual percentage rate	2.85%

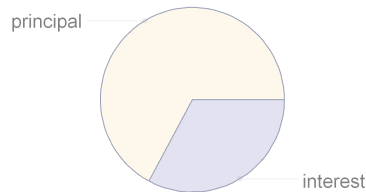
Monthly payments:

monthly payment	Kč16542
effective interest rate	2.888%

Units

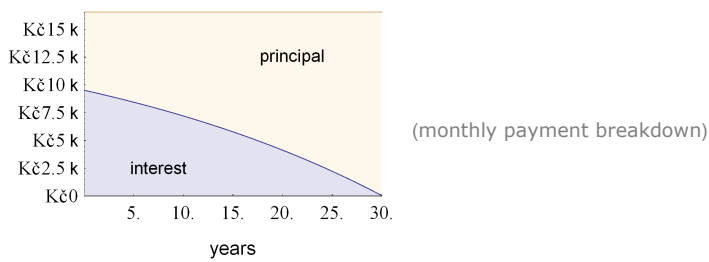
Mortgage totals:

principal paid	Kč4 million
total interest paid	Kč1.955 million
total payments	Kč5.955 million



Units

Payments and balances:



Out[55]=

```
In[56]:= WolframAlpha["fixed rate mortgage",
  IncludePods -> {"PaymentsTable"}, AppearanceElements -> {"Pods"},
  InputAssumptions -> {"*F.FixedRateMortgage.MA-_4000000+Czech+koruny",
    "*F.FixedRateMortgage.MP-_30+yr", "*F.FixedRateMortgage.APR-_2.85+%25"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Payments table:

year	monthly payment	ending balance	yearly principal paid	yearly interest paid
1	Kč16 542	Kč3.914 million	Kč85 620	Kč112 887
2	Kč16 542	Kč3.826 million	Kč88 093	Kč110 415
3	Kč16 542	Kč3.736 million	Kč90 636	Kč107 871
4	Kč16 542	Kč3.642 million	Kč93 253	Kč105 254
5	Kč16 542	Kč3.546 million	Kč95 946	Kč102 561
6	Kč16 542	Kč3.448 million	Kč98 717	Kč99 791
7	Kč16 542	Kč3.346 million	Kč101 567	Kč96 941
8	Kč16 542	Kč3.242 million	Kč104 500	Kč94 008
9	Kč16 542	Kč3.134 million	Kč107 517	Kč90 990
10	Kč16 542	Kč3.024 million	Kč110 622	Kč87 886
11	Kč16 542	Kč2.91 million	Kč113 816	Kč84 692
12	Kč16 542	Kč2.793 million	Kč117 103	Kč81 405
13	Kč16 542	Kč2.672 million	Kč120 484	Kč78 024
14	Kč16 542	Kč2.548 million	Kč123 963	Kč74 545
15	Kč16 542	Kč2.421 million	Kč127 542	Kč70 965
16	Kč16 542	Kč2.289 million	Kč131 225	Kč67 282
17	Kč16 542	Kč2.154 million	Kč135 014	Kč63 493
18	Kč16 542	Kč2.015 million	Kč138 913	Kč59 595
19	Kč16 542	Kč1.873 million	Kč142 924	Kč55 584
20	Kč16 542	Kč1.725 million	Kč147 051	Kč51 457
21	Kč16 542	Kč1.574 million	Kč151 297	Kč47 210
22	Kč16 542	Kč1.419 million	Kč155 666	Kč42 842
23	Kč16 542	Kč1.258 million	Kč160 161	Kč38 347
24	Kč16 542	Kč1.094 million	Kč164 785	Kč33 722
25	Kč16 542	Kč924 042	Kč169 544	Kč28 964
26	Kč16 542	Kč749 602	Kč174 439	Kč24 068

Out[56]=

27	Kč16 542	Kč570 126	Kč179 476	Kč19 031
28	Kč16 542	Kč385 467	Kč184 659	Kč13 849
29	Kč16 542	Kč195 477	Kč189 991	Kč8 517
30	Kč16 542	Kč0	Kč195 477	Kč3 031

[+ Units](#)

WolframAlpha dále řeší zmíněnou problematiku aktuálních měnových kurzů. Následující příklad demonstruje vývoj kurzu eura vůči dolaru.

```
In[57]:= WolframAlpha["euro - dollar exchange rate",
  IncludePods -> {"Input", "Result", "MakeChangeMoreThanOneCoin:QuantityData",
    "LocalCurrencyConversion", "History"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

convert €1 (euro) to US dollars

Result:

\$1.13 (US dollars)

[More accuracy](#)

Minimal currency form:

[Show physical characteristics](#)

[Show non-metric](#)



(total coin weight: 9.768 grams)

Local currency conversion:

Kč27.75 (Czech koruny) (at current quoted rate)

Out[57]=

Exchange history for €1 (euro):

[Reverse rate](#)

[Last year](#) | ▾



1-year minimum	\$1.11 (26. 1. 2015 1 day ago)
1-year maximum	\$1.39 (19. 3. 2014 10 months ago)
1-year average	\$1.31 (annualized volatility: 5.8%)

```
In[58]:= WolframAlpha["euro - dollar exchange rate", IncludePods -> {"CurrencyConversions"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[58]=

Additional currency conversions for €1 (euro): World currencies Reverse rates

JPY	¥133.6 (Japanese yen)
GBP	74.93p (British pence)
CNY	¥ 7.06 (Chinese yuan)
CAD	C\$1.41 (Canadian dollars)
INR	₹69.48 (Indian rupees)
RUB	py676.55 (Russian rubles)
CHF	CHF1.03 (Swiss francs)

Poslední příklad popisuje vývoj hodnoty dolaru vzhledem k inflaci od roku 1980. Jedním z výstupů je grafické zobrazení tohoto vývoje a průměrná hodnota inflace za rok.

```
In[59]:= WolframAlpha["$10,000 in 1980", IncludePods ->
  {"Input", "Result", "History", "AverageRateOfInflation", "TotalRateOfInflation"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"History__Log scale", "History__Linear scale"}]
```

Out[59]=

Input interpretation:
\$10000 (current US dollars) in 1980

Result:
\$3423.60 (1980 US dollars)
(based on Consumer Price Index)

History: Log scale

(from 1980 to 2015)
(in US dollars)

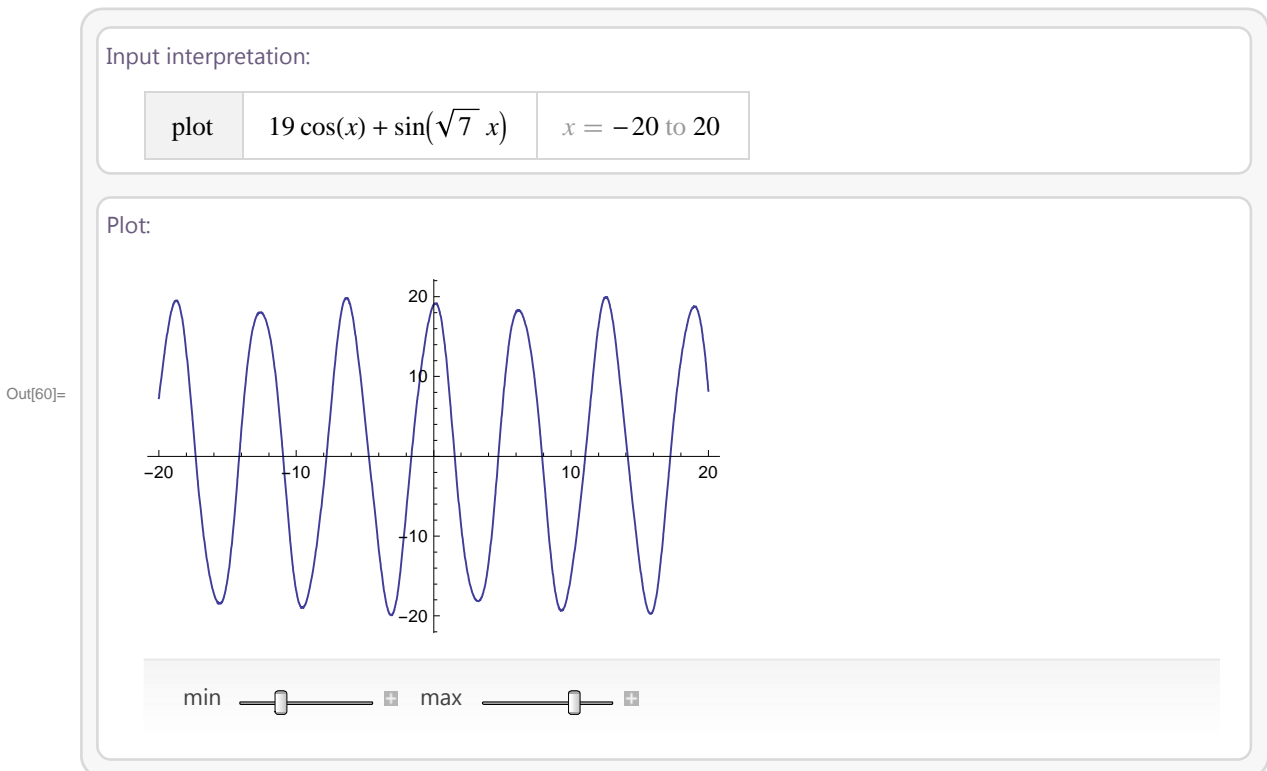
Average rate of inflation:
3.11% per year

Total inflation factor:
192.1%

5.3 Grafy v kartézské soustavě

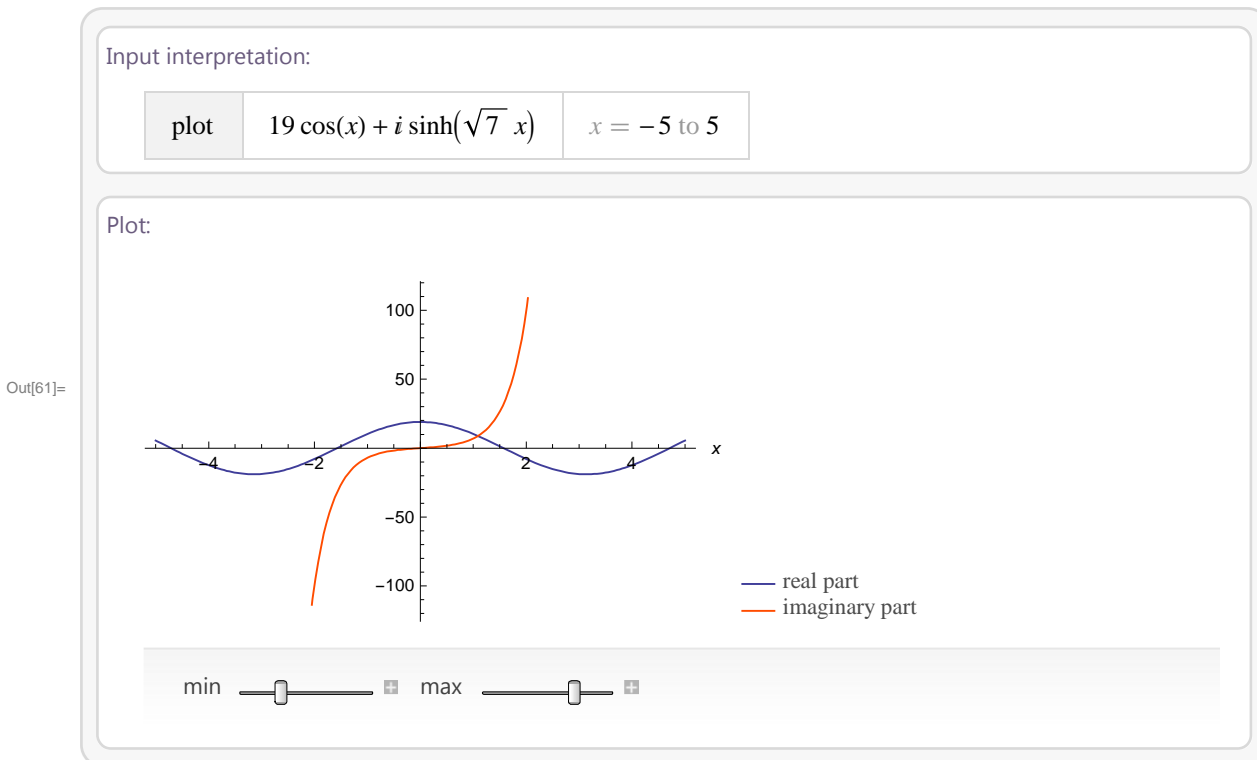
Jedním ze silných nástrojů sw *Mathematica*, ale také WolframAlpha je práce s grafy v 2D a 3D rozměru. Během několika sekund po zadání příkazu obdržíme velmi pěkné grafy, které lze dále velmi jednoduše upravovat pro naše potřeby. Např. po zadání “*plot sin(sqrt(7)x) + 19 cos(x) between -20 and 20*” jednoduše získáme graf v 2D.

```
In[60]:= WolframAlpha["plot sin(sqrt(7)x) + 19 cos(x) between -20 and 20",
  IncludePods -> {"Input", "Plot"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```



Tento graf jednoduše upravíme změnou několika parametrů, např. místo $\sqrt{7}$, napíšeme $-\sqrt{7}$ a meze grafu změníme od -5 do 5 .

```
In[61]:= WolframAlpha["plot sin(sqrt(-7)x) + 19 cos(x) between -5 and 5",
  IncludePods -> {"Input", "Plots"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```



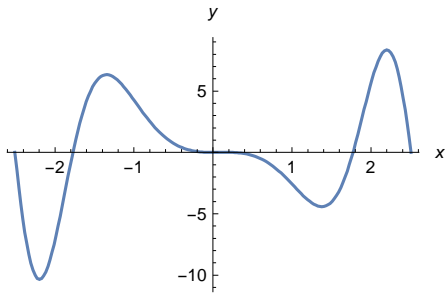
V obou grafech jsme zadali rozsah grafu. Co se ovšem stane, když rozsah nezádáme? Např. zadáním příkazu “*plot (1 - 4x - x^3/17) sin(x^2)*”, vidíme, že WolframAlpha zobrazí dva grafy, každý s jiným rozsahem. V dalším příkladě “*plot 2 x + 1, 1 - x^2, 1 - x - x^2/3*” vidíme, že zobrazený rozsah grafu hraje důležitou roli. Proto je důležité mít znalosti z matematiky při zadávání příkazů v *sw Mathematica* nebo WolframAlpha.


```
In[62]:= WolframAlpha["plot (1 - 4x - x^3/17) sin(x^2)", IncludePods -> {"Input", "Plot"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

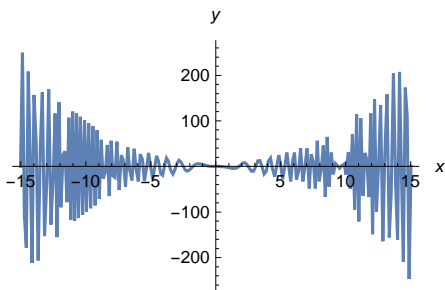
Input:

$$\left(1 - 4x - \frac{x^3}{17}\right) \sin(x^2)$$

Plots:



min max



min max

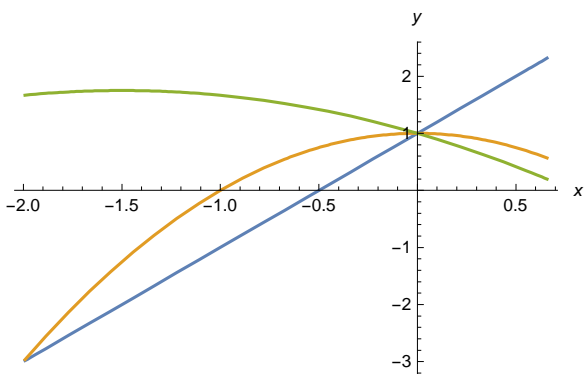
Out[62]=

```
In[63]:= WolframAlpha["plot 2 x + 1, 1 - x^2, 1 - x - x^2/3", IncludePods -> {"Input", "Plot"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

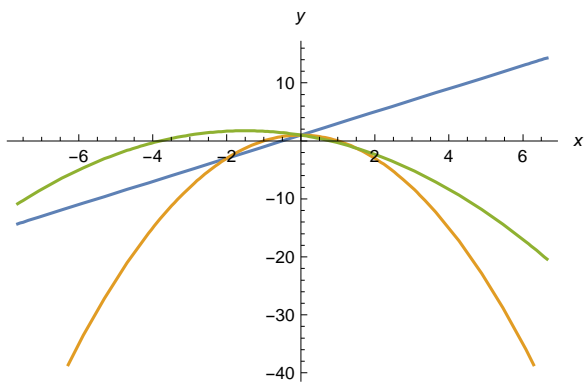
	$2x + 1$
plot	$1 - x^2$
	$1 - x - \frac{x^2}{3}$

Plots:



Out[63]=

min max



min max

Stejně jako grafy v 2D, je jednoduché vykreslit hezké a názorné grafy ve WolframAlpha v 3D.

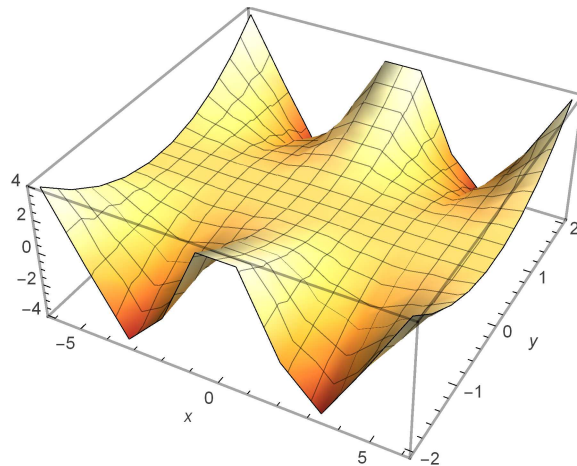
```
In[64]:= WolframAlpha["plot y^2 cos(x), x=-6..6, y=-2..2", IncludePods -> {"Input", "3DPlot"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Input interpretation:

plot	$y^2 \cos(x)$	$x = -6$ to 6
		$y = -2$ to 2

3D plot:

Out[64]=



Pokud do WolframAlpha zadáme více než jednu funkci, např. “plot sin(x) cos(y), cos(x y)”, poté se vykreslí každá z funkcí zvlášť a jestliže nezadáme konkrétní rozsah grafů, WolframAlpha, zvolí nejvhodnější rozsah pro každý z grafů.

```
In[65]:= WolframAlpha["plot sin(x) cos(y), cos(x y)", IncludePods -> {"Input"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

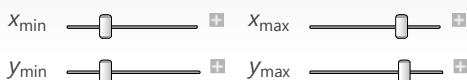
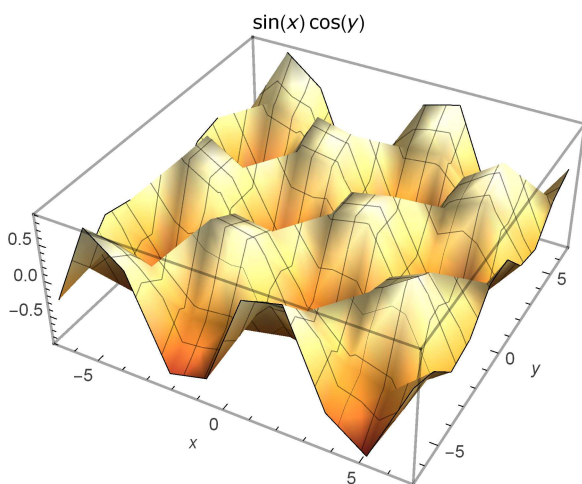
Input interpretation:

Out[65]=

plot	$\sin(x) \cos(y)$
	$\cos(x y)$

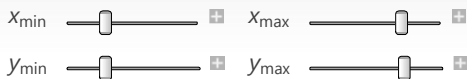
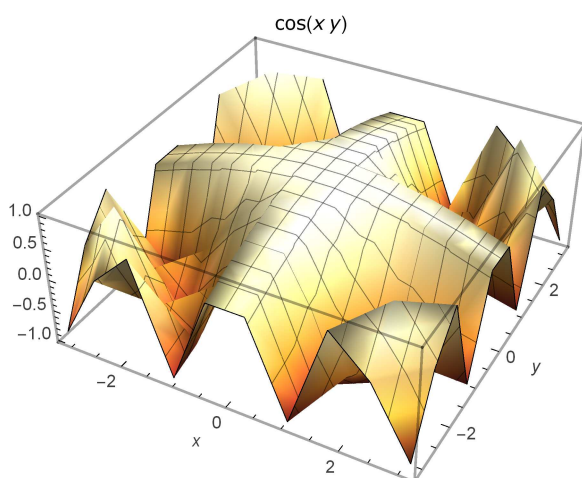
```
In[66]:= WolframAlpha["plot sin(x) cos(y), cos(x y)", IncludePods -> {"3DPlot"},
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

3D plots:



More controls

Out[66]=



More controls

5.4 Řešení příkladů krok za krokem

Velkou výhodou WolframAlpha při řešení složitějších matematických příkladů, je možnost zobrazení nejenom správného výsledku, ale i podrobného postupu, jak k tomuto řešení dojít. Např. zadáním příkazu “*solve* ($9^{(x+1)} - (28(3^{(x)})) + 3 == 0$ over the real numbers” vidíme nejenom výsledek, ale i celý postup řešení. Tato možnost umožňuje studentům lepší pochopení

probíraného učiva a učitelům přípravu zajímavých příkladů do výuky. Výhodné je použít tento celkový popis řešení při řešení limit např. “*limit of $(x - 3) / (x^2 - 2x - 3)$ as x approaches 3*” nebo derivací např. “*derivative of $x^4 + 9x^3 + 7x - 2$* ”.

```
In[67]:= WolframAlpha["solve (9^(x + 1)) - (28 (3^(x))) + 3 == 0 over the real numbers",
  IncludePods -> {"Input", "Result", "RootPlot"}, AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"Result__Step-by-step solution"}]
```

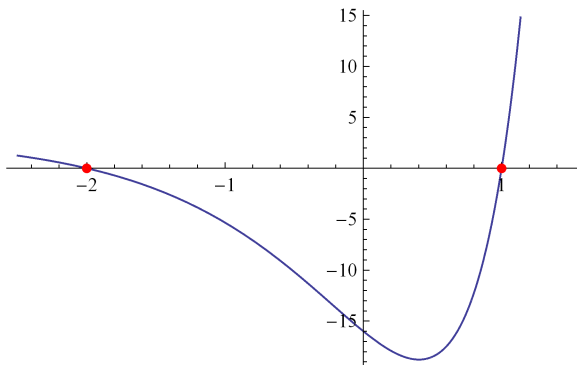
Input interpretation:

solve

$$9^{x+1} - 28 \times 3^x + 3 = 0$$

over the reals

Root plot:



Out[67]=

```
In[68]:= WolframAlpha["limit of (x - 3) / (x^2 - 2x - 3) as x approaches 3",
  IncludePods -> "Limit", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"Limit__Step-by-step solution"}]
```

Out[68]=

Limit: Approximate form Hide steps

$$\lim_{x \rightarrow 3} \frac{x - 3}{x^2 - 2x - 3} = \frac{1}{4}$$

Possible intermediate steps:

Find the following limit:

$$\lim_{x \rightarrow 3} \frac{x - 3}{x^2 - 2x - 3}$$

Factor the numerator and denominator:

$$\lim_{x \rightarrow 3} \frac{x - 3}{(x - 3)(x + 1)}$$

Cancel terms, assuming $x - 3 \neq 0$:

$$\lim_{x \rightarrow 3} \frac{1}{x + 1}$$

Applying the quotient rule, write $\lim_{x \rightarrow 3} \frac{1}{x + 1}$ as $\frac{\lim_{x \rightarrow 3} 1}{\lim_{x \rightarrow 3} (x + 1)}$:

$$\frac{1}{\lim_{x \rightarrow 3} (x + 1)}$$

$x + 1$ is a polynomial and thus everywhere continuous, so

$$\lim_{x \rightarrow 3} (x + 1) = 1 + 3 = 4:$$

Answer:

$$\frac{1}{4}$$

ℝ is the set of real numbers »

```
In[69]:= WolframAlpha["derivative of x^4 + 9x^3 + 7x - 2", IncludePods -> "Input",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"Input__Step-by-step solution"}]
```

Derivative:

Hide steps

$$\frac{d}{dx}(x^4 + 9x^3 + 7x - 2) = 4x^3 + 27x^2 + 7$$

Possible intermediate steps:

Possible derivation:

$$\frac{d}{dx}(x^4 + 9x^3 + 7x - 2)$$

Differentiate the sum term by term and factor out constants:

$$= \frac{d}{dx}(-2) + 7\left(\frac{d}{dx}(x)\right) + 9\left(\frac{d}{dx}(x^3)\right) + \frac{d}{dx}(x^4)$$

The derivative of -2 is zero:

$$= 7\left(\frac{d}{dx}(x)\right) + 9\left(\frac{d}{dx}(x^3)\right) + \frac{d}{dx}(x^4) + \boxed{0}$$

Simplify the expression:

$$= 7\left(\frac{d}{dx}(x)\right) + 9\left(\frac{d}{dx}(x^3)\right) + \frac{d}{dx}(x^4)$$

The derivative of x is 1:

$$= 9\left(\frac{d}{dx}(x^3)\right) + \frac{d}{dx}(x^4) + \boxed{1} 7$$

Use the power rule, $\frac{d}{dx}(x^n) = n x^{n-1}$, where $n = 3$: $\frac{d}{dx}(x^3) = 3x^2$:

$$= 7 + \frac{d}{dx}(x^4) + 9 \boxed{3x^2}$$

Simplify the expression:

$$= 7 + 27x^2 + \frac{d}{dx}(x^4)$$

Use the power rule, $\frac{d}{dx}(x^n) = n x^{n-1}$, where $n = 4$: $\frac{d}{dx}(x^4) = 4x^3$:

Answer:

$$= 7 + 27x^2 + \boxed{4x^3}$$

WolframAlpha nám zobrazí postup řešení neurčitých integrálů. Například řešení neurčitého integrálu $\int x^2 \sin x \, dx$ pomocí metody per partes nebo neurčitých integrálů $\int x e^{x^2} \, dx$ a $\int \sin x \cos^2 x \, dx$ metodou substituce.

```
In[70]:= WolframAlpha["Integrate x^2 sin x", IncludePods -> "IndefiniteIntegral",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
  PodStates -> {"IndefiniteIntegral__Step-by-step solution"}]
```

Indefinite integrals:

Hide steps

$$\int x^2 \sin(x) \, dx = 2x \sin(x) - (x^2 - 2) \cos(x) + \text{constant}$$

Possible intermediate steps:

Take the integral:

$$\int x^2 \sin(x) \, dx$$

For the integrand $x^2 \sin(x)$, integrate by parts, $\int f \, dg = f g - \int g \, df$, where

$$f = x^2, \quad dg = \sin(x) \, dx,$$

$$df = 2x \, dx, \quad g = -\cos(x):$$

$$= -x^2 \cos(x) + 2 \int x \cos(x) \, dx$$

For the integrand $x \cos(x)$, integrate by parts, $\int f \, dg = f g - \int g \, df$, where

$$f = x, \quad dg = \cos(x) \, dx,$$

$$df = dx, \quad g = \sin(x):$$

$$= -x^2 \cos(x) + 2x \sin(x) - 2 \int \sin(x) \, dx$$

The integral of $\sin(x)$ is $-\cos(x)$:

$$= x^2 (-\cos(x)) + 2x \sin(x) + 2 \cos(x) + \text{constant}$$

Which is equal to:

Answer:

$$= 2x \sin(x) - (x^2 - 2) \cos(x) + \text{constant}$$

Out[70]=


```
In[71]:= WolframAlpha["Integrate x exp x^2", IncludePods -> "IndefiniteIntegral",
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic},
PodStates -> {"IndefiniteIntegral__Step-by-step solution"}]
```

Indefinite integrals:

Approximate form

Hide steps

$$\int x \exp(x^2) dx = \frac{e^{x^2}}{2} + \text{constant}$$

Possible intermediate steps:

Take the integral:

$$\int e^{x^2} x dx$$

For the integrand $e^{x^2} x$, substitute $u = x^2$ and $du = 2 x dx$:

$$= \frac{1}{2} \int e^u du$$

The integral of e^u is e^u :

$$= \frac{e^u}{2} + \text{constant}$$

Substitute back for $u = x^2$:

Answer:

$$= \frac{e^{x^2}}{2} + \text{constant}$$

Out[71]=

```
In[72]:= WolframAlpha["integrate sin(x)cos(x)^2", IncludePods -> "IndefiniteIntegral",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic},
  PodStates -> {"IndefiniteIntegral__Step-by-step solution"}]
```

Indefinite integrals:

Hide steps

$$\int \sin(x) \cos^2(x) dx = -\frac{1}{3} \cos^3(x) + \text{constant}$$

Possible intermediate steps:

Take the integral:

$$\int \sin(x) \cos^2(x) dx$$

For the integrand $\sin(x) \cos^2(x)$, substitute $u = \cos(x)$ and $du = -\sin(x) dx$:

$$= -\int u^2 du$$

The integral of u^2 is $\frac{u^3}{3}$:

$$= -\frac{u^3}{3} + \text{constant}$$

Substitute back for $u = \cos(x)$:

Answer:

$$= -\frac{1}{3} \cos^3(x) + \text{constant}$$

Out[72]=

V případě určitých integrálů WolframAlpha zobrazí kromě výsledku také jeho geometrický význam. Přínosem je také dynamický výstup závislosti integrálního součtu na dělení intervalu.

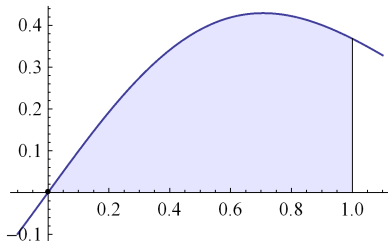
```
In[73]:= WolframAlpha["Integrate x exp -x^2 from 0 to 1", IncludePods ->
{"Input", "VisualRepresentationOfTheIntegral", "RiemannSums", "IndefiniteIntegral"},
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Definite integral:

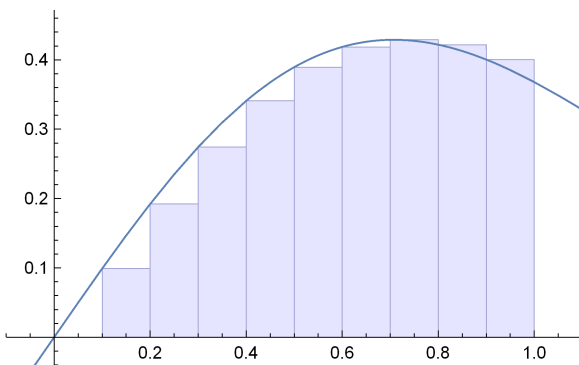
[More digits](#)[Step-by-step solution](#)

$$\int_0^1 x \exp(-x^2) dx = \frac{e-1}{2e} \approx 0.31606$$

Visual representation of the integral:



Riemann sums:



$$\text{integral: } \frac{e-1}{2e} \approx 0.31606$$

Riemann sum: 0.296525

error: 0.0195352

number of subintervals



summation method

left endpoint

midpoint

right endpoint

Indefinite integral:

[Approximate form](#)[Step-by-step solution](#)

$$\int x \exp(-x^2) dx = -\frac{e^{-x^2}}{2} + \text{constant}$$

Metodou krok za krokem lze zobrazit řešení diferenciálních rovnic např. “ $y'(t) - 2y(t) = 3e^{2t}$ ”.

```
In[74]:= WolframAlpha["y'(t) - 2y(t) = 3 e^(2t)", IncludePods -> "DifferentialEquationSolution",
AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic},
PodStates -> {"DifferentialEquationSolution__Step-by-step solution"}]
```

Differential equation solutions:

Approximate form

Solve as a linear equation | ▾

Hide steps

$$y(t) = c_1 e^{2t} + 3 e^{2t} t$$

Possible intermediate steps:

Solve the linear equation $\frac{dy(t)}{dt} - 2y(t) = 3e^{2t}$:

Let $\mu(t) = e^{\int -2 dt} = e^{-2t}$.

Multiply both sides by $\mu(t)$:

$$e^{-2t} \frac{dy(t)}{dt} - (2e^{-2t})y(t) = 3$$

Substitute $-2e^{-2t} = \frac{d}{dt}(e^{-2t})$:

$$e^{-2t} \frac{dy(t)}{dt} + \frac{d}{dt}(e^{-2t})y(t) = 3$$

Apply the reverse product rule $g \frac{df}{dt} + f \frac{dg}{dt} = \frac{d}{dt}(fg)$ to the left- hand side:

$$\frac{d}{dt}(e^{-2t} y(t)) = 3$$

Integrate both sides with respect to t :

$$\int \frac{d}{dt}(e^{-2t} y(t)) dt = \int 3 dt$$

Evaluate the integrals:

$e^{-2t} y(t) = 3t + c_1$, where c_1 is an arbitrary constant.

Divide both sides by $\mu(t) = e^{-2t}$:

Answer:

$$y(t) = e^{2t} (3t + c_1)$$

Out[74]=

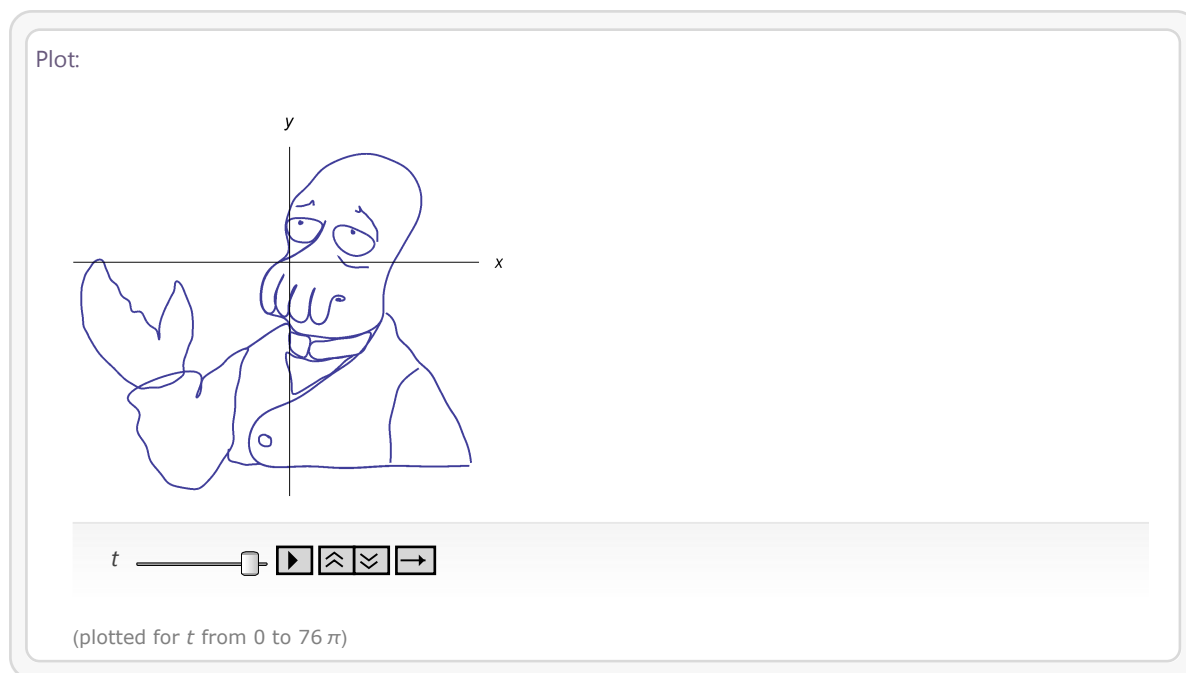
5.5 Vizualizace rovnic

WolframAlpha nemusí nabízet pouze spousty odpovědí na naše často složité otázky, nebo řešení složitých matematických příkladů, chemických rovnic, či geografických údajů. WolframAlpha může ovšem také sloužit k zábavě a pobavení. Jednou z hezkých ukázek, toho co vše WolframAlpha dokáže, je zobrazení velmi pěkných obrázků z křivek, které jsou zadány ve formě matematických rovnic. Např. zadáním příkazu “*butterfly curve*” vidíme obrázek motýla a parametrickou rovnici, z které je obrázek vykreslen. Podobných zajímavých křivek lze ve WolframAlpha vykreslit mnohem více (např. “*Zoidberg-like curve*”, “*Obama-like curve*”, “*Luxo Jr.-like curve*”).

Pozn. Parametrické rovnice, které matematicky popisují zobrazené křivky, nejsou z důvodu dlouhého kódu (3-4 stránky) uvedeny v papírové verzi této publikace.

```
In[75]= WolframAlpha["Zoidberg-like curve", IncludePods -> "PlotPod:PopularCurveData",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

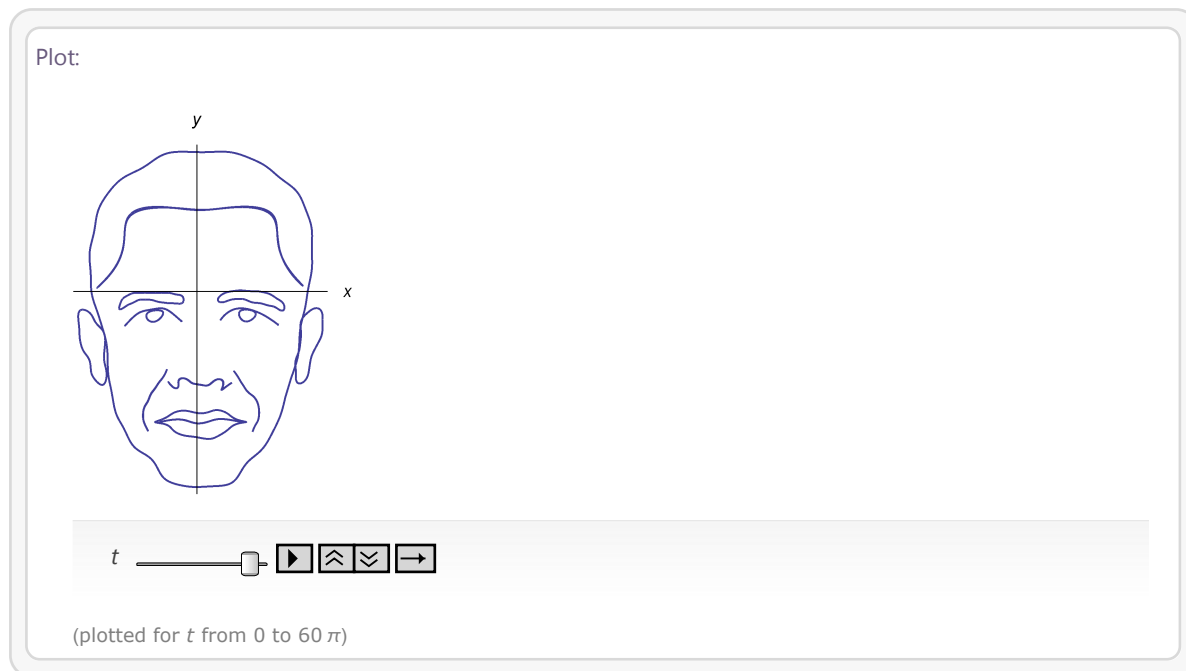
Out[75]=



```
In[76]= WolframAlpha["Zoidberg-like curve",
  IncludePods -> "EquationsPod:PopularCurveData", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}];
```

```
In[77]:= WolframAlpha["Obama-like curve", IncludePods -> "PlotPod:PopularCurveData",
  AppearanceElements -> {"Pods"}, TimeConstraint -> {30, Automatic, Automatic, Automatic}]
```

Out[77]=



```
In[78]:= WolframAlpha["Obama-like curve",
  IncludePods -> "EquationsPod:PopularCurveData", AppearanceElements -> {"Pods"},
  TimeConstraint -> {30, Automatic, Automatic, Automatic}];
```

6 Závěr

Anotace

Publikace je věnována možnostem, které nabízí systém WolframAlpha, jež přinesla v roce 2009 společnost Wolfram Research. Cílem publikace není obecný návod, jak WolframAlpha používat. WolframAlpha se poměrně rychle vyvíjí a obecný návod zřejmě ani podat nelze. Přínos systému WolframAlpha je ukázán na příkladech ze širokého spektra oborů - matematice, fyzice, chemii, astronomii, historii, kultuře, ekonomii nebo metrologii. Příklady jsou vybrány tak, aby z nich bylo patrné, jakým způsobem lze zadat hledanou úlohu a jaké výsledky dokáže WolframAlpha najít.

Závěr

WolframAlpha lze charakterizovat jako odpovídací stroj, tzv. “computational knowledge engine” vytvořený společností Wolfram Research. Jedná se o webovou službu, která se snaží odpovídat na dotazy uživatelů, na rozdíl od vyhledávacích služeb, které obvykle poskytují pouze seznam stránek pravděpodobně obsahujících odpověď. WolframAlpha, který byl spuštěn v květnu 2009, je založen na dlouhodobě vyvíjeném produktu společnosti Wolfram Research Mathematica, což je silný nástroj pro technické výpočty používaný matematiky, fyziky, analytiky a mnoha dalšími odborníky z řady jiných oborů, který obsahuje rozsáhlé možnosti pro řešení řady úloh, které zahrnují např. řešení rovnic, programování, import a export dat nebo vizualizaci funkcí a dat. Později (v únoru 2012) byl spuštěn WolframAlpha Pro, který nabízí uživatelům doplňkové služby za měsíční poplatek. Hlavní službou je zde možnost stažení mnoha typů souborů a dat, mezi něž patří zejména tabulky typu raw, obrázky, zvukové soubory, soubory XML nebo mnoho specializovaných formátů pro využití ve vědě, medicíně a matematice. Dalšími výhodami jsou rozšířená klávesnice, interaktivita s CDF soubory, možnost úprav a uložení grafických a datových výstupů, zobrazení řešení “krok za krokem”, a také více výpočetního času.

Summary

WolframAlpha is a computational knowledge engine or an answer engine developed by Wolfram Research. It is an online service that answers factual queries directly by computing the answer from externally sourced data rather than providing a list of documents or web pages that might contain the answer as a search engine might. WolframAlpha, which was released on May, 2009, is based on Wolfram's earlier flagship product Mathematica, a tool for technical computing that is used by mathematicians, engineers, analysts, and many others around the world. Mathematica contains an extensive knowledge base for working with a very broad range of tasks, including solving equations, programming, importing and exporting data, visualizing functions and data, and so on.

Later on (February, 2012), WolframAlpha Pro was released offering users additional features for a monthly subscription fee. A key feature is the ability to upload many common file types and data including raw tabular data, images, audio, XML, and dozens of specialized scientific, medical, and mathematical formats for automatic analysis. The other features include an extended keyboard, interactivity with CDF, data downloads, in-depth step by step solution, the ability to customize and save graphical and tabular results and extra computation time.

7 Použitá literatura

Homepage of Wolfram Research. URL: <http://wolfram.com/>.

WolframAlpha. URL: <http://wolframalpha.com/>.

WolframAlpha Blog. URL: <http://blog.wolframalpha.com/>.

Mgr. Jan Říha, Ph.D. a kolektiv

WolframAlpha ve výuce přírodovědných a ekonomických předmětů

Výkonný redaktor Prof. RNDr. Zdeněk Dvořák, DrSc., Ph.D.

Odpovědná redaktorka Mgr. Jana Kreiselová

Technická redakce autor

Určeno pro studenty a akademické pracovníky VŠ

Vydala a vytiskla Univerzita Palackého v Olomouci

Křížkovského 8, 771 47 Olomouc

www.upol.cz/vup

e-mail: vup@upol.cz

Olomouc 2015

1. vydání

z. č. 2015/0078

Edice - Odborné publikace

ISBN 978-80-244-4471-0

Neprodejná publikace