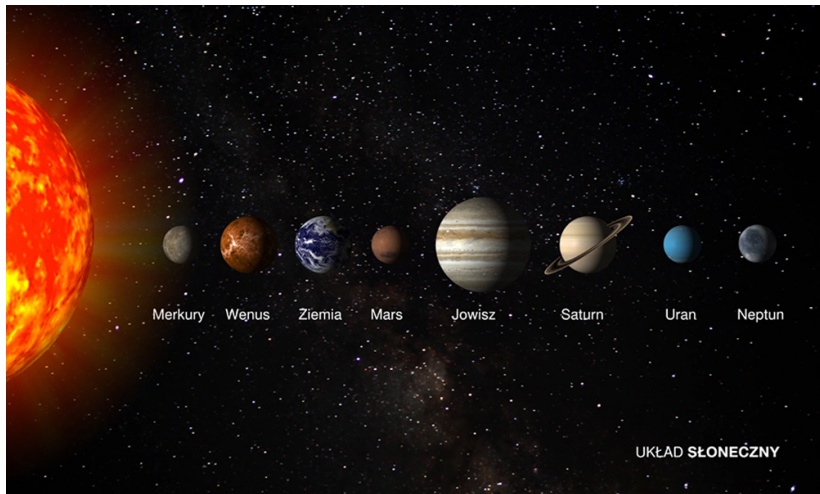


Program obliczający odległość wybranej planety od Ziemi

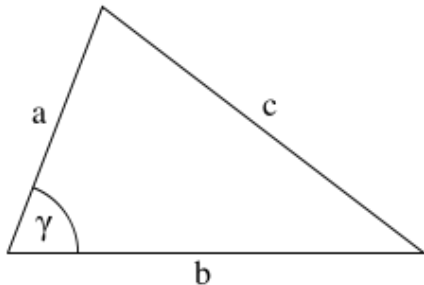
Ewa Czerewko
Józef Borsuk

18 stycznia 2017

Układ Słoneczny



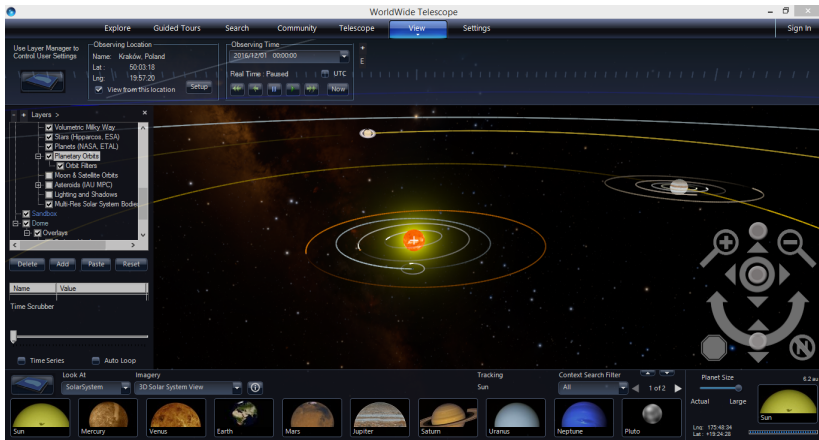
Twierdzenie cosinusów



$$c^2 = a^2 + b^2 - ab \cos(\gamma)$$

$$\gamma = \gamma_0 + \omega t$$

WorldWide Telescope



Średnie prędkości planet

Merkury	$47.36 \frac{km}{s}$
Wenus	$35.02 \frac{km}{s}$
Ziemia	$29.78 \frac{km}{s}$
Mars	$24.13 \frac{km}{s}$
Jowisz	$13.07 \frac{km}{s}$
Saturn	$9.64 \frac{km}{s}$
Uran	$6.80 \frac{km}{s}$
Neptun	$5.43 \frac{km}{s}$

Średnie odległości planet od Słońca

Merkury	57 909 050 km
Wenus	108 208 926 km
Ziemia	149 598 261 km
Mars	227 936 637 km
Jowisz	778 412 020 km
Saturn	1 426 725 413 km
Uran	2 870 972 220 km
Neptun	4 495 600 000 km

Średnia prędkość kątowna

$$\omega = \frac{v}{r}$$

```

*plik3.scm (-) - gedit
(define ny_time (current-time))

(define a 1480550400)
(define t ((lambda(x y) (- x y)) ny_time a))
(define (funkcja r1 r2 alfa1 alfa2)
  (sqrt(+ (expt r1 2) (- (expt r2 2) (* 2 r1 r2 (cos (- alfa1 alfa2)))))))

(define rme 57909050)
(define rw 108208926)
(define rz 149598261)
(define rma 227936637)
(define rjo 778412020)
(define rsa 1426725413)
(define rur 2870972220)
(define rne 4495600000)

(define alfaw (+ (* 246 (/ 3.141593 180)) (* t 8.18 (expt 10 -7))))
(define alfaw (+ (* 292 (/ 3.141593 180)) (* t 3.24 (expt 10 -7))))
(define alfaz (+ (* t 1.99 (expt 10 -7))))
(define alfama (+ (* 288 (/ 3.141593 180)) (* t 1.06 (expt 10 -7))))
(define alfaja (+ (* 119 (/ 3.141593 180)) (* t 1.68 (expt 10 -8))))
(define alfasa (+ (* 189 (/ 3.141593 180)) (* t 6.75 (expt 10 -9))))
(define alfaur (+ (* 315 (/ 3.141593 180)) (* t 2.37 (expt 10 -9))))
(define alfane (+ (* 272 (/ 3.141593 180)) (* t 1.21 (expt 10 -9))))

(define L '1-Merkury 2-Wenus 3-Mars 4-Jowisz 5-Saturn 6-Uran 7-Neptun)
(display "Wpisz numer planety, ktorej odleglosc chcesz sprawdzic\n")
(print L)
(display "\n")
(define (prompt/read prompt)
  (display prompt)
  (read))

```



```

*plik3.scm (-) - gedit
[Icons] Open Save Undo Cut Copy Paste Find
+plik3.scm x
(define L '(1-Merkury 2-Wenus 3-Mars 4-Jowisz 5-Saturn 6-Uran 7-Neptun))
(display "Wpisz numer planety, ktorej odleglosc chcesz sprawdzic\n\n")
(print L)
(display "\n")
(define (prompt/read prompt)
  (display prompt)
  (read-line))

(define r (prompt/read "Twój wybór: "))

(define d1 (funkcja rz rme alfaz alfame))
(define d2 (funkcja rz rw alfaz alfaw))
(define d3 (funkcja rz rma alfaz alfama))
(define d4 (funkcja rz rjo alfaz alfajjo))
(define d5 (funkcja rz rsa alfaz alfasa))
(define d6 (funkcja rz rur alfaz alfaur))
(define d7 (funkcja rz rne alfaz alfane))

(if (eq? (string->number r) 1)
  (begin (print 'aktualna 'odleglosc: d1 'kn)
         (print 'Minimalna 'odleglosc: (- rz rme) 'km 'Maksymalna 'odleglosc: (+ rz rme) 'kn)
         (if (<= d1 (sqrt(+ (expt rz 2) (expt rme 2))))
             (display "Dobry moment na obserwacje!\n"))))

(if (eq? (string->number r) 2)
  (begin (print 'aktualna 'odleglosc: d2 'kn)
         (print 'Minimalna 'odleglosc: (- rz rw) 'km 'Maksymalna 'odleglosc: (+ rz rw) 'kn)
         (if (<= d2 (sqrt(+ (expt rz 2) (expt rw 2))))
             (display "Dobry moment na obserwacje!\n"))))

Scheme Tab Width: 8 Ln 30, Col 29 INS

```

Dziękujemy za uwagę!