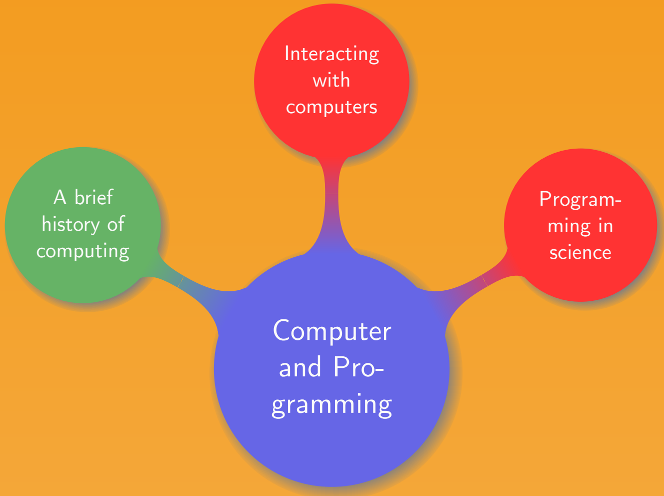
Introduction to Computer and Programming

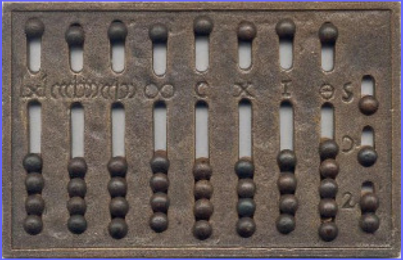
1. Computer and Programming

Manuel – Summer 2019

Chapter organisation



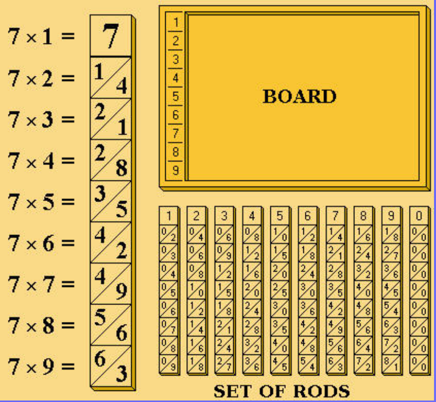
Ancient era



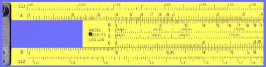
Abacus (-2700)

Antikythera mechanism (-100)

Calculation tools

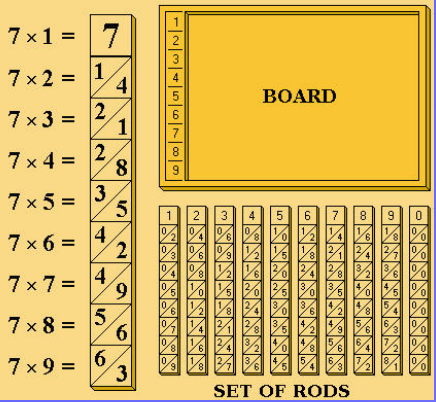


Napier's bones (1617)

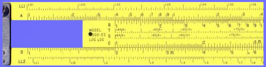


Sliderule (1620)

Calculation tools



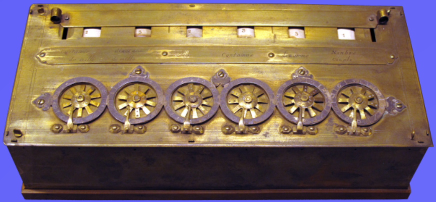
Napier's bones (1617)



Sliderule (1620)

First pocket calculator introduced around 1970 in Japan

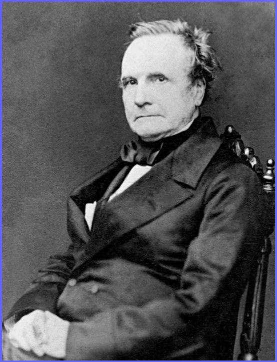
Mechanical calculators



Pascaline (1642)

Arithmomètre (1820)

The 19th century



Charles Babbage (1791–1871) achievements:

- Difference engine: built in the 1990es
- Analytical engine: never built

Ada Byron (1815–1852) achievements:

- Extensive notes on Babbage's engines
- Algorithm to calculate Bernoulli numbers

The birth of modern computing

First part of the 20th century:

- 1936: First freely programmable computer
- 1946: First electronic general-purpose computer

- 1936: First freely programmable computer
- 1948: Invention of the transistor
- 1951: First commercial computer
- 1958: Integrated circuit

UNIVAC I (1951)

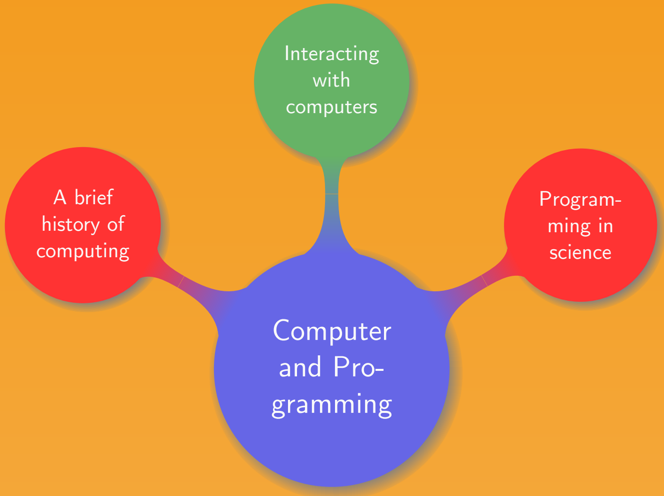
Modern computing

Apple I (1976)

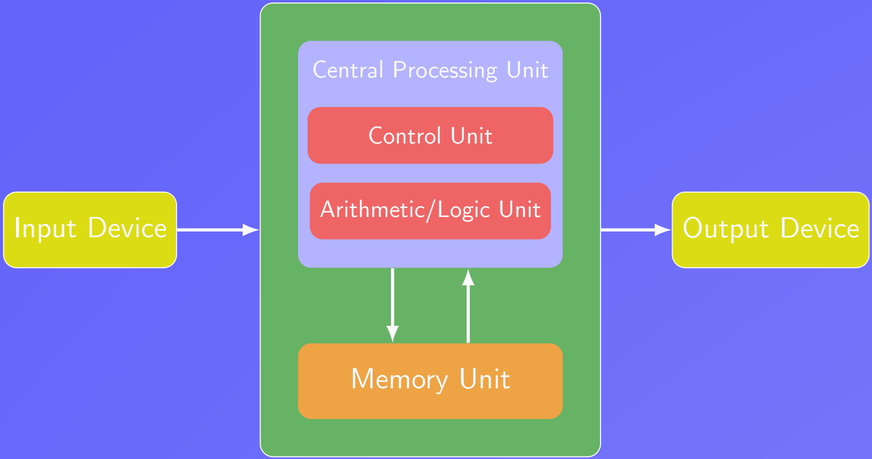
Second part of the 20th century:

- 1962: First computer game
- 1969: ARPAnet
- 1971: First microprocessor
- 1975: First consumer computers
- 1981: First PC, MS-DOS
- 1983: First home computer with a GUI
- 1985: Microsoft Windows
- 1991: Linux

Chapter organisation



Von Neumann architecture



What does a computer understand?

Numbers in various bases:

- Humans use *decimal* (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), e.g. $(253)_{10}$
- Computers work internally using *binary* (0,1), e.g $(11111101)_2$
- Human-friendly way to represent binary: *hexadecimal* (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F), e.g. $(FD)_{16}$

Number base conversion

Base conversion:

- From base b into decimal: evaluate the polynomial
  $(11111101)_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 253$

  $(\mathrm{FD})_{16} = \mathrm{F} \cdot 16^1 + \mathrm{D} \cdot 16^0 = 15 \cdot 16^1 + 13 \cdot 16^0 = 253$

- From decimal into base b: repeatedly divide n by b until the quotient is 0. Consider the remainders from right to left
  rem(253,2)=1, rem(126,2)=0, rem(63,2)=1, rem(31,2)=1, rem(15,2)=1, rem(7,2)=1, rem(3,2)=1, rem(1,2)=1

  rem(253,16)=13=D, rem(15,16)=15=F

- From base b into base b^a: group numbers into chunks of a elements

  $(11111101)_2 = 1111\ 1101 = (\mathrm{FD})_{16}$

Quick examples

Exercise.

- Convert into hexadecimal: 1675, 321, $(100011)_2$, $(10111011)_2$
- Convert into binary: 654, 2049, ACE, 5F3EC6
- Convert into decimal: $(111110)_2$, $(10101)_2$, $(12345)_{16}$, 12C3C

Quick examples

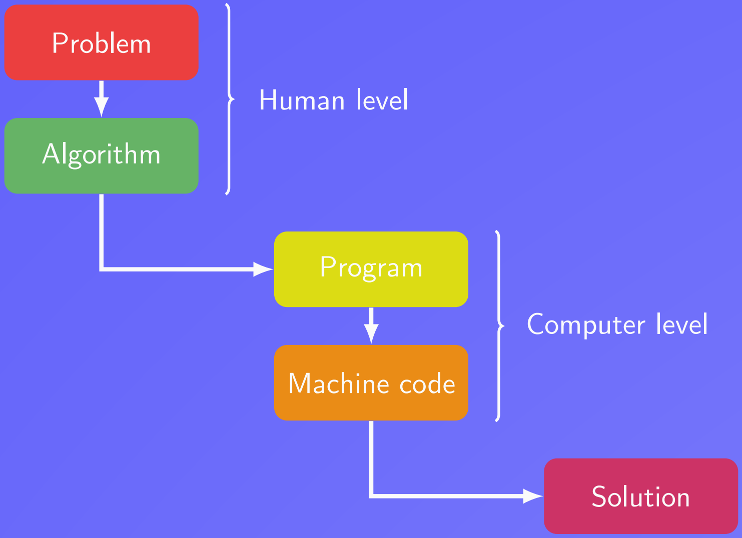
Exercise.

- Convert into hexadecimal: 1675, 321, $(100011)_2$, $(10111011)_2$
- Convert into binary: 654, 2049, ACE, 5F3EC6
- Convert into decimal: $(111110)_2$, $(10101)_2$, $(12345)_{16}$, 12C3C

Solution.

- $1675 = (68B)_{16}$, $321 = (141)_{16}$, $(100011)_2 = (23)_{16}$
- $654 = 1010001110$, $2049 = 100000000001$,
  $\mathrm{ACE} = 101011001110$, $\mathrm{5F3EC6} = 10111110011111011000110$
- $(111110)_2 = 62$, $(10101)_2 = 21$, $(12345)_{16} = 74565$,
  $\mathrm{12C3C} = 76860$

How to use a computer?



Algorithm

Algorithm: recipe explaining the computer how to solve a problem

Algorithm

Algorithm: recipe explaining the computer how to solve a problem

Example.
Detail an algorithm to prepare a jam sandwich.

Actions: `cut`, `listen`, `spread`, `sleep`, `take`, `eat`, `dip`, `assemble`
Things: `knife`, `guitar`, `bread`, `honey`, `jam jar`, `sword`, `slice`

Algorithm

Algorithm: recipe explaining the computer how to solve a problem

Example.
Detail an algorithm to prepare a jam sandwich.

Actions: `cut`, `listen`, `spread`, `sleep`, `take`, `eat`, `dip`, `assemble`
Things: `knife`, `guitar`, `bread`, `honey`, `jam jar`, `sword`, `slice`

Algorithm. (*Sandwich making*)

Input : 1 bread, 1 jam jar, 1 knife
Output: 1 jam sandwich

take the knife and cut 2 slices of bread;
dip the knife into the jam jar;
spread the jam on the bread, using the knife;
assemble the 2 slices together, jam on the inside;

Humans and computers

From algorithm to machine code

Algorithm → Programming language

↓

Machine code ← Interpreter or compiler

A first program

Example.

Given a square and the length of one side, what is its area?

Algorithm.

Input : side (the length of one side of a square)

Output: the area of the square

return side $\times$ side

A first program

Example.
Given a square and the length of one side, what is its area?

Algorithm.

Input : side (the length of one side of a square)
Output: the area of the square
return side $\times$ side

To obtain the result in MATLAB:

1. Type the code
2. Press `Enter`

area.m

```
a=input("Side: ");
printf ("Area: %d",  a*a)
```

Running the program

area.c

```c
#include<stdio.h>
int main() {
  int side;
  printf("Side: ");
  scanf("%d",&side);
  printf("Area: %d", side*side);
  retrun 0;
}
```

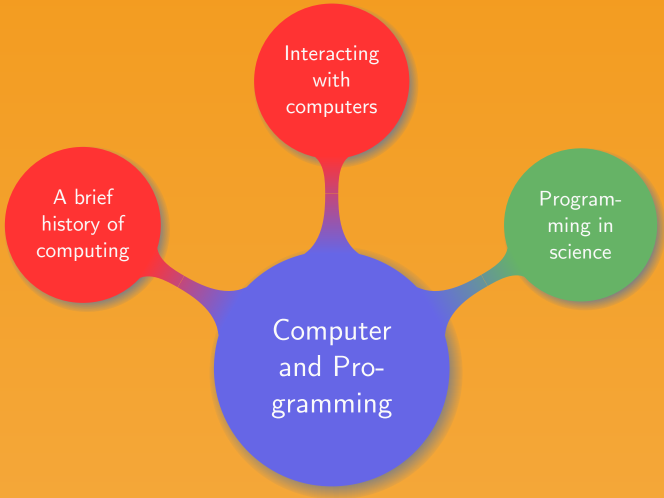
area.cpp

```cpp
#include <iostream>
using namespace std;
int main() {
int side;
  cout << "Side: ";  cin >> side;
  cout << "Area: " << side*side;
  return 0;
}
```

To obtain the result in C or C++

1. Write the source code
2. Compile the program
3. Run the program

Chapter organisation



MATLAB

Common mathematics software:

- Axiom
- GAP
- GP/PARI
- Magma
- Maple
- MATLAB
- Maxima
- Octave
- R
- Scilab
- Mathematica

[1]Studied in VG101

MATLAB

Common mathematics software:

- Axiom
- GAP
- GP/PARI
- Magma
- Maple
- MATLAB
- Maxima
- Octave
- R
- Scilab
- Mathematica

MATrix LABoratory (MATLAB):

- Matrix manipulations[1]
- Implement algorithms[1]
- Plotting functions and data[1]
- User interface creation

[1] Studied in VG101

MATLAB

Common mathematics software:

- Axiom
- GAP
- GP/PARI
- Magma
- Maple
- MATLAB
- Maxima
- Octave
- R
- Scilab
- Mathematica

MATrix LABoratory (MATLAB):

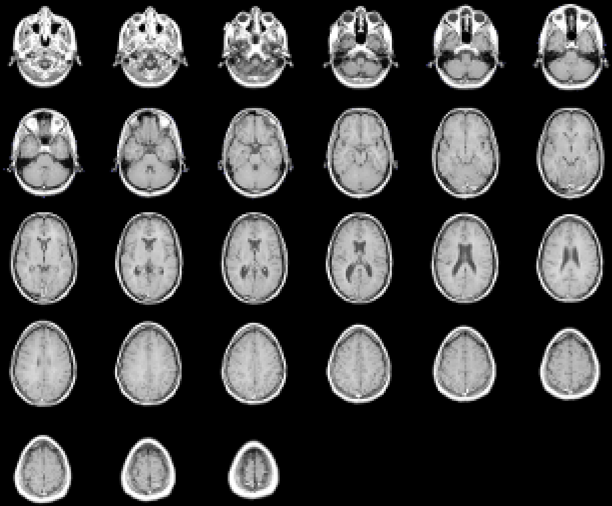
- Matrix manipulations[1]
- Implement algorithms[1]
- Plotting functions and data[1]
- User interface creation

Benefits of MATLAB:

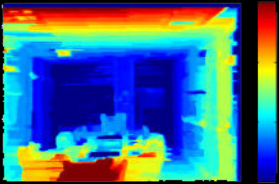
- Easy to use
- Built-in language
- Versatile
- Many toolboxes

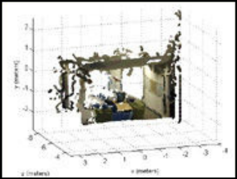
[1] Studied in VG101

MRI slices

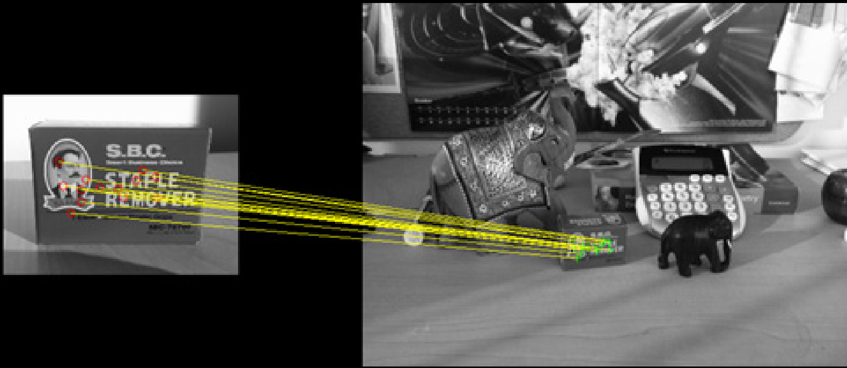


Stereo Vision





Object detection



Mathematics and Physics

$$\begin{tikzcd}
\text{Physics problem} \arrow[r] & \text{Mathematics problem} \arrow[d] \\
& \text{Mathematical solution}
\end{tikzcd}$$

Mathematics and Physics

$$
\begin{tikzcd}
\text{Physics problem} \arrow[r] & \text{Mathematics problem} \arrow[d] \\
\text{Physical solution} & \text{Mathematical solution} \arrow[l]
\end{tikzcd}
$$

Mathematics and Physics

$$\begin{tikzcd}
\text{Physics problem} \arrow[r] & \text{Mathematics problem} \arrow[d] \\
\text{Physical solution} \arrow[u] & \text{Mathematical solution} \arrow[l]
\end{tikzcd}$$

What to do?

Before jumping on the computer and starting to code:

- Clearly state or translate the problem
- Define what is known as the *input*
- Define what is to be found as the *output*
- Develop an *algorithm*, i.e. a systematic way to solve the problem
- Verify the solution on simple input
- Implementing the algorithm

Solving a problem using a computer

Example.
Given that the sun is located $1.496 \cdot 10^8$ km away from the Earth and has a circumference of $4.379 \cdot 10^6$ km, calculate its density.

Solving a problem using a computer

Example.
Given that the sun is located $1.496 \cdot 10^8$ km away from the Earth and has a circumference of $4.379 \cdot 10^6$ km, calculate its density.

Strategy to solve the problem:

- Easy part
  - Problem: finding the density of the sun
  - Input: distance r, circumference c
  - Output: density d

Solving a problem using a computer

Example.
Given that the sun is located $1.496 \cdot 10^8$ km away from the Earth and has a circumference of $4.379 \cdot 10^6$ km, calculate its density.

Strategy to solve the problem:

- Easy part
  - Problem: finding the density of the sun
  - Input: distance r, circumference c
  - Output: density d
- Finding the density is slightly more complicated:
  1. Approximate the Sun by a sphere and determine its volume V
  2. Think of Kepler's third law $\frac{T^2}{r^3} = \frac{4\pi^2}{GM}$
  3. Apply Kepler's third law to find the mass $M = \frac{4\pi^2 r^3}{GT^2}$

The Algorithm

Algorithm. (*Desnity of the Sun*)

Input : $r = 1.496 \cdot 10^8,\ c = 4.379 \cdot 10^6,\ G = 6.674 \cdot 10^{-11},$
$T = 365$

Output: Density of the Sun

$V \leftarrow \frac{4}{3}\pi(\frac{c}{2\pi})^3$;
$M \leftarrow \frac{4\pi^2 r^3}{GT^2}$;
return $\frac{M}{V}$;

The Algorithm

Algorithm. (*Desnity of the Sun*)

Input : $r = 1.496 \cdot 10^8,\ c = 4.379 \cdot 10^6,\ G = 6.674 \cdot 10^{-11},$
$T = 365$

Output: Density of the Sun

1 $V \leftarrow \frac{4}{3}\pi(\frac{c}{2\pi})^3$;

2 $M \leftarrow \frac{4\pi^2 r^3}{GT^2}$;

3 **return** $\frac{M}{V}$;

After running the algorithm we find 338110866080

WRONG!

WRONG!

Units are not consistent...

The Algorithm

Algorithm. (*Density of the Sun*)

Input : $r = 1.496 \cdot 10^{11}$ m, $c = 4.379 \cdot 10^{9}$ m, $T = 365 * 24 * 3600$ s, $G = 6.674 \cdot 10^{-11}$ m^3/kg/s^2

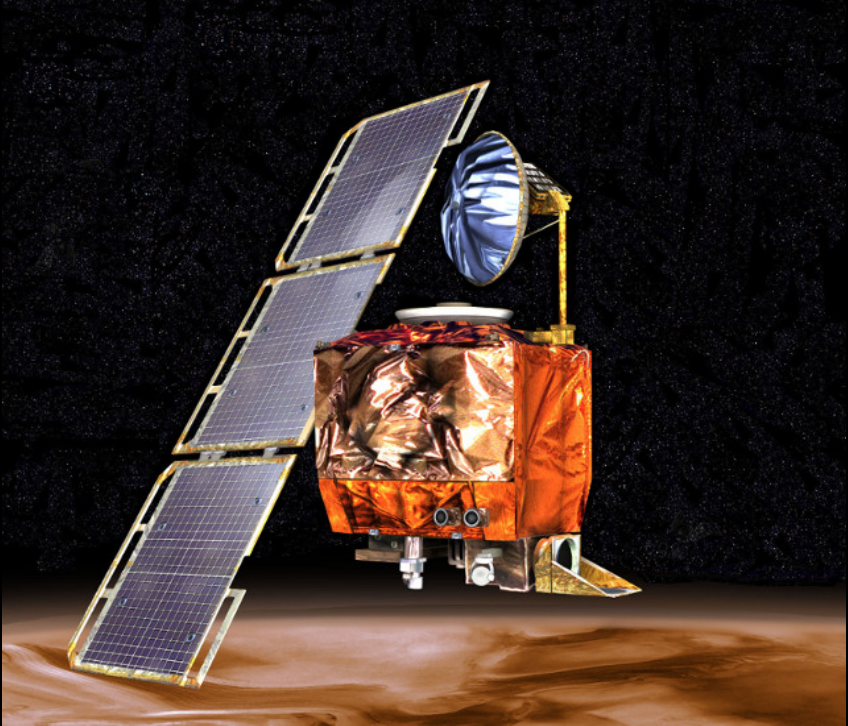
Output: Density of the Sun

$V \leftarrow \frac{4}{3}\pi(\frac{c}{2\pi})^3$;

$M \leftarrow \frac{4\pi^2 r^3}{GT^2}$;

return $\frac{M}{V}$;

After running the algorithm we find 1404 kg/m^3



Key points

- What is a programming language?
- What are the two main types of programming language?
- What is an algorithm?
- How to tackle a problem?

Thank you!

References I

1.3 https://upload.wikimedia.org/wikipedia/commons/b/b5/RomanAbacusRecon.jpg

1.3 https://upload.wikimedia.org/wikipedia/commons/7/76/Antikythera_model_front_panel_Mogi_Vicentini_2007.JPG

1.4 https://upload.wikimedia.org/wikipedia/commons/5/54/Batons_de_Napier.png

1.4 https://upload.wikimedia.org/wikipedia/commons/a/a0/Pocket_slide_rule.jpg

1.5 https://upload.wikimedia.org/wikipedia/commons/7/78/Pascaline-CnAM_823-1-IMG_1506-black.jpg

1.5 https://upload.wikimedia.org/wikipedia/commons/8/83/Thomas_Arithmometer_1975.png

1.6 https://upload.wikimedia.org/wikipedia/commons/6/6b/Charles_Babbage_-_1860.jpg

1.6 https://upload.wikimedia.org/wikipedia/commons/b/b1/Ada_Byron_aged_seventeen_%281832%29.jpg

1.7 http://www.cftea.com/c/2011/02/WPRMMSW80E5HFKI7/97GZAM05GBMCV7P9.jpg

1.8 https://upload.wikimedia.org/wikipedia/commons/a/a1/Apple_I_Computer.jpg

1.21 MATLAB documentation

1.22 MATLAB documentation

1.23 MATLAB documentation

References II

1.30 https://upload.wikimedia.org/wikipedia/commons/thumb/1/19/Mars_Climate_Orbiter_2.jpg/660px-Mars_Climate_Orbiter_2.jpg