

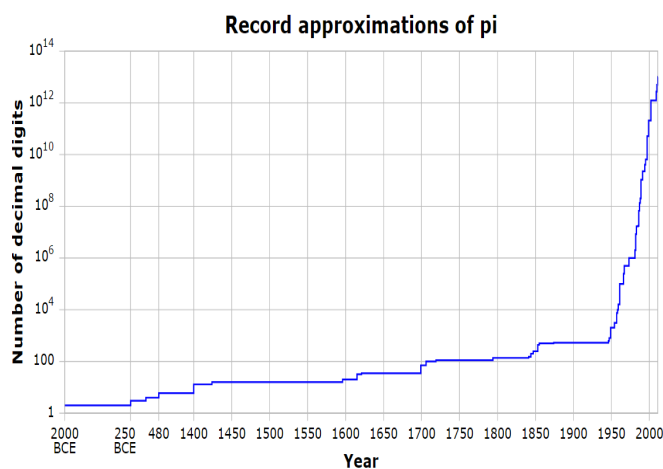
MTH 371: Group Project 1

Taylor Series and Digits of π

GENERAL GROUP PROJECT GUIDELINES:

- Group project assignments should be a collaborative effort. All should participate in discussion and solution writing.
- Two weeks after the project is assigned, your group will meet with Dr. Vidden to discuss. All members must be present. Your grade will be determined at the end of the meeting.
- Each student should keep group project solutions in a dedicated notebook. Bring this notebook to your weekly meeting to discuss your findings. For coded solutions, bring a laptop to your weekly meeting. Have the laptop ready before the start of the meeting.

For thousands of years, many people have tried to compute digits of π to the highest possible precision. Below is a record of the number of digits computed over history. Notice the dramatic increase following the advent of the computer following World War II. See the Wikipedia page (http://en.wikipedia.org/wiki/Approximations_of_%CF%80) for some of the story regarding the hunt for digits of π .



1. Taylor series are a major tool used to calculate digits of π , specifically, the following series for arctangent.

$$\arctan(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{2n+1}, \quad R = 1, \quad IoC = [-1, 1]$$

Derive this series for arctangent by using a geometric series. Explain why the radius of convergence is $R = 1$. Explain why the interval of convergence is $[-1, 1]$.

Hint: For the interval of convergence, you need to check the endpoints separately using a series test from Calculus II.

2. Using the arctangent Taylor series from the previous problem, one can create a series approximation for π from the following three formulas.

$$\frac{\pi}{4} = \arctan(1) \quad (\text{leads to Leibniz's formula for } \pi) \quad (1)$$

$$\frac{\pi}{4} = \arctan\left(\frac{1}{2}\right) + \arctan\left(\frac{1}{3}\right) \quad (\text{leads to Euler's formula for } \pi) \quad (2)$$

$$\frac{\pi}{4} = 4 \arctan\left(\frac{1}{5}\right) - \arctan\left(\frac{1}{239}\right) \quad (\text{leads to Machin's formula for } \pi) \quad (3)$$

There are countless others which can be created in this way.

- (a) Verify the identity (1).
 - (b) Using the Taylor series for arctangent, program each of the three formulas in Scilab. Compare and contrast the performance of each. How fast are they? How many terms of the series are needed for exact digits of π ? Find the absolute error at each step. The first is rather slow. What do you think the reason is?
3. With problem 2, you are limited to machine precision (about 16 decimal digits in Scilab). So how do you get more digits of π ? I have attached a script written in Matlab which accomplished this task.
- (a) Translate this script into Scilab and run it.
 - (b) Give a complete explanation as to how the script works. You should understand every line of code and the role it plays. This script uses the same strategy as was used with the very first π computation via a computer as accomplished by Jon von Neumann in 1949. See the paper on D2L for the story of this 1949 computation.
 - (c) Compare each of the three formulas from problem 2 using your new many digit program.
4. OPTIONAL challenge problem. Derive the last two formulas given in problem 2 using some trigonometric fantasticness.

```
function s = machin0(m)
%MACHIN Calculate decimals of pi with Machin's formula
%   MACHIN(M) gives a string with pi truncated to M decimals
%   using Machin's formula:  $\pi = 16*\text{acot}(5) - 4*\text{acot}(239)$ 

%   Author: Jonas Lundgren <splinefit@gmail.com> 2008

if nargin < 1, m = 50; end
base = 1e15;
n = ceil(m/15) + 1;

% Machin's formula
x = xacot(5,16,n,base) - xacot(239,4,n,base);

% Canonical form
carry = 0;
for k = n+1:-1:1
    xk = x(k) + carry;
    carry = floor(xk/base);
    x(k) = xk - carry*base;
end

% Write string
s = sprintf('%15.0f',x(2:n));
s(isspace(s)) = '0';
s = ['3.', s(1:m)];

%-----
function y = xacot(a,c,n,base)
% Calculate C*ACOT(A) to N digits in base BASE

b = a*a;
m = floor(n*log(base)/log(b));
```

```
x = [c; zeros(n,1)];           % x = c
y = 0;                         % y = 0
for k = m:-1:0
    y = xdiv(x,2*k+1,base) - xdiv(y,b,base);   % y = c/✓
(2*k+1) - y/b
end
y = xdiv(y,a,base);           % y = y/a
```

```
%-----
```

```
function x = xdiv(x,a,base)
% Calculate X/A to N digits in base BASE
```

```
r = 0;
t = floor(base/a);
u = base - a*t;
for k = 1:numel(x)
    v = x(k) + r*u;
    q = floor(v/a);
    x(k) = q + r*t;
    r = v - a*q;
end
```