Analysis of Beam using FINITE ELEMENT METHOD (in SMath Studio platform)

NOTE:
This worksheet requires that the associated Plugin (Structural Beam Analysis Utility Functions by Redem Legaspi Jr.) be downloaded and enabled from SMath Studio Extension Manager tool.

DATA INPUT SIMPLE GUIDE:

a.) BEAM DIMENSION AND SUPPORT DATA:

1.) Beam Length Input Data

BEAM LENGTH DATA may take any desired 'distance unit', either in METRIC or in 'IMPERIAL' system. Unit system COMPATIBILITY is internally taken care of by the program. Hence, assigning any distance unit such as feet, meter, yard, mile, kilometer, cm, mm, inch, etc -- is absolutely permitted.

Beam Length Data:

$L_b := 40\ ft$

2.) Support Input Data

Support Input Data adapts the following syntax in 2-row matrix format:

Syntax: $\begin{bmatrix} S1 & S2 & S3 & ... & Sn \\ X1 & X2 & X3 & ... & Xn \end{bmatrix}$

Where:

S1,S2,S3... Sn are STRINGS (quoted words) that specify the type of support assigned at the given locations along the beam.

Valid SUPPORT TYPES are as follows:

a.) "PIN" or "PINNED" for pin support
b.) "FIX" or "FIXED" for fixed support
c.) "ROLLER" for roller support

Note that in this application, ROLLER and PIN supports are treated as technically the same.

X1, X2, X3... Xn are floating point values that correspond to the LOCATION of individual support. Each value MUST be associated with DISTANCE UNITS either in english or in metric system (e.g,. meter, ft, cm). An infinite number of supports maybe assigned to the beam; there is no limit so long as the computer's computing capacity can still handle the complexity of the resulting input matrices.

LOCATION is defined as the distance from the left end of the beam to any point along the beam -- which is, by default, set at the ORIGIN or ZERO distance.

Support Data:

$$Supports := \begin{bmatrix} \text{"FIXED"} & \text{"PIN"} & \text{"PIN"} \\ 0\ ft & 20\ ft & 40\ ft \end{bmatrix}$$

b.) BEAM LOAD INPUT DATA:

Input data for 'Beam Loads' is specially formatted in a logical manner that is understood by SMath Studio. Like MS Excel software, SMath Studio is a special application that reads data (such that an input data should be written in a proper syntax or format), does calculations and returns results.

The syntax of the input data is a programmer's discretion (and is decided based mainly on how Smath Studio paper-like interface picks and reads data) -- which for this application is defined in matrix format as below:

1.) Concentrated Load Data adapts the following syntax in 2-row matrix format:

```
Syntax: ┌                                    ┐
        │  P1   P2   P3   ...   Pn │
        │  X1   X2   X3   ...   Xn │
        └                                    ┘
```

Where:

P1,P2,P3... Pn are floating point values of concentrated loads. The units can be either in METRIC, ENGLISH or MIXED system (e.g., kip, kN, lbf).

X1,X2,X3... Xn are the locations of concentrated loads measured from the origin or the left end of the beam. Units can be either in METRIC or ENGLISH system (e.g,. meter, ft, cm). Mixed unit system is also allowed.

Concentrated Load Data:

$$pLoad := \begin{bmatrix} 20\ kip & 10\ kip & 20\ kip \\ 2\ ft & 6\ ft & 30\ ft \end{bmatrix}$$

2.) Moment Load Data adapts the following syntax in 2-row matrix format:

```
Syntax: ┌                                    ┐
        │  M1   M2   M3   ...   Mn │
        │  X1   X2   X3   ...   Xn │
        └                                    ┘
```

Where:

M1,M2,M3... Mn are floating point values of moment loads. The units can be either in METRIC, ENGLISH or MIXED system (e.g., kip-ft, kN-m, kN-ft, etc).

X1,X2,X3... Xn are the locations of moment loads measured from the origin or the left end of the beam. Units can be either in METRIC or ENGLISH system (e.g,. meter, ft, cm). Mixed unit system is also allowed.

Moment Load Data:

$$mLoad := \begin{bmatrix} 10\ kip\ ft & 15\ kip\ ft \\ 2\ m & 20\ ft \end{bmatrix}$$

3.) Trapezoidal Load Data adapts the following syntax in 4-row matrix format:

Syntax:

$$\begin{bmatrix} Ws1 & Ws2 & Ws3 & ... & Wsn \\ We1 & We2 & We3 & ... & Wen \\ Xs1 & Xs2 & Xs3 & ... & Xsn \\ Xe1 & Xe2 & Xe3 & ... & Xen \end{bmatrix}$$

Where:

Ws1,Ws2,Ws3... Wsn are floating point START values of trapezoidal loads.

We1,We2,We3... Wen are floating point END values of trapezoidal loads. The units can be either in METRIC, ENGLISH or MIXED system (e.g., kip/ft, kN/m, kN/ft, etc).

Xs1,Xs2,Xs3... Xsn are the locations of START values of trapezoidal loads measured from the origin or the left end of beam.

Xe1,Xe2,Xe3... Xen, on the other hand, are the locations of END values of trapezoidal loads also measured from the origin or the left end of the beam.

Units can be either in METRIC or ENGLISH system. Mixed unit system is also allowed (e.g,. meter, ft, cm).

Trapezoidal Load Data:

$$tLoad := \begin{bmatrix} 5\ \frac{kip}{ft} & 15\ \frac{kip}{ft} \\ 10\ \frac{kip}{ft} & 5\ \frac{kip}{ft} \\ 8\ ft & 35\ ft \\ 12\ ft & 40\ ft \end{bmatrix}$$

4.) Uniform Load data adapts the following syntax in 3-row matrix format:

Syntax:

$$\begin{bmatrix} W1 & W2 & W3 & ... & Wn \\ Xs1 & Xs2 & Xs3 & ... & Xsn \\ Xe1 & Xe2 & Xe3 & ... & Xen \end{bmatrix}$$

Where:

W1,W2,W3... Wn are floating point values of uniform loads. The units can be w1,w2,w3... wn are floating point values of uniform loads. The units can be either in METRIC, ENGLISH or MIXED system (e.g., kip/ft, kN/m, kN/ft, etc).

Xs1,Xs2,Xs3... Xsn are the start locations of uniform loads measured from the origin or the left end of the beam.

Xe1,Xe2,Xe3... Xen, on the other hand, are the end locations of uniform loads also measured from the origin or the left end of the beam.

Units can be either in METRIC or ENGLISH system. Mixed unit system is also allowed (e.g,. meter, ft, cm).

Uniform Load Data:

$$uLoad := \begin{bmatrix} 15\ \frac{kip}{ft} & 10\ \frac{kip}{ft} \\ 15\ ft & 25\ ft \\ 22\ ft & 28\ ft \end{bmatrix}$$

c.) BEAM GEOMETRIC AND DESIGN DATA:

1.) Modulus of Elasticity data adapts the following syntax in 3-row matrix format:

```
         ┌                                               ┐
         │  E1     E2     E3      ...    En   |
Syntax:  │  Xs1    Xs2    Xs3   ...    Xsn |
         │  Xe1    Xe2    Xe3   ...    Xen |
         └                                               ┘
```

Where:

E1,E2,E3... En are floating point values of moduli of elesticity. The units can be either in METRIC, ENGLISH or MIXED system (e.g., ksi, MPa, psf, psi, etc).

Xs1,Xs2,Xs3... Xsn are the start locations of moduli of elasticity measured from the origin or the left end of the beam.

Xe1,Xe2,Xe3... Xen, on the other hand, are the end locations of moduli of elasticity also measured from the origin or the left end of the beam.

Units can be either in METRIC or ENGLISH system. Mixed unit system is also allowed (e.g,. meter, ft, cm).

Modulus of Elasticity:

$$EData := \begin{bmatrix} 29000\ ksi & 28000\ ksi \\ 0\ ft & 20\ ft \\ 20\ ft & L_b \end{bmatrix}$$

Note:
This example assumes that the beam is composed of two segments with different moduli of elasticity.

2.) Moments of Inertia data adapt the following syntax in 3-row matrix format:

```
Syntax:  [ I1     I2     I3     ...    In  ]
         [ Xs1    Xs2    Xs3   ...    Xsn ]
         [ Xe1    Xe2    Xe3   ...    Xen ]
```

Where:

I1,I2,I3... In are floating point values of moments of inertia. The units can be either in METRIC, ENGLISH or MIXED system (e.g,. m^4, ft^4, in^4, cm^4, mm^4, etc).

Xs1,Xs2,Xs3... Xsn are the start locations of moments of inertia measured from the origin or the left end of the beam.

Xe1,Xe2,Xe3... Xen, on the other hand, are the end locations of moments of inertia also measured from the origin or the left end of the beam.

Units can be either in METRIC or ENGLISH system. Mixed unit system is also allowed (e.g,. meter, ft, cm).

Moment of Inertia of Beam Segment-1:

$I_{x1} := 23300\ in^4$

Moment of Inertia of Beam Segment-2:

$I_{x2} := 19600\ in^4$

Moment of Inertia Data:

$$IData := \begin{bmatrix} I_{x1} & I_{x2} \\ 0\ ft & 20\ ft \\ 20\ ft & L_b \end{bmatrix}$$

Note:
This example assumes that the beam is composed of two segments with different moments of inertia.

d.) PARSING DATA TO GRAPHICS MATRIX:

Before 'BEAM', 'BEAM LOADS' and 'SUPPORTS' data can be shown graphically or converted to drawing, each data should first undergo Data Parsing. The purpose of which is to convert each data into a matrix format that can be plotted onto SMath Studio 2D Chart.

This part is what makes SMath Studio plugin programming challenging, interesting and enjoyable on the part of programming enthusiasts...

1.) Parse 'Beam Data' into Graphics Matrix Format:

Syntax: ParseBeam [BmSpan, BmThk]

Where:

BmSpan = Span of Beam
BmThk = Apparent Beam Thickness/Plot Scale

Apparent Beam Thickness/Plot Scale:

$bt := 0.5$

Parse Beam Span for Graphics Output:

$plot\_beam := \text{ParseBeam}\left(L_b, bt\right)$

2.) Parse 'Support Data' into Graphics Matrix Format:

Syntax: ParseSupport [SptMtxData, BmSpan, Prefix, BmThk, Scale]

Where:
SptMtxData = Support data definition in Matrix format
BmSpan = Beam span
Prefix = Support label prefix
BmThk = Beam apparent thickness
Scale = Graphics plot scale

Supports Plot Scale:

$sScale := 1.5$

Parse Support Data for Graphics Output:

$plot\_support := \text{ParseSupport}\left(Supports, L_b, \text{"R"}, bt, sScale\right)$

3.) Parse 'Concentrated Load Data' into Graphics Matrix Format:

Syntax: ParsePLoad [BmPLoad, Prefix, Scale]

Where:
BmPLoad = Concentrated load data definition in Matrix format
Prefix = Concentrated load label prefix
Scale = Graphics plot scale

Concentrated Load Plot Scale:

$pScale := 5$

Parse Concentrated Load Data for Graphics Output:

$plot\_pLoad := \text{ParsePLoad}\left(pLoad, \text{"P"}, pScale\right)$

4.) Parse 'Moment Load Data' into Graphics Matrix Format:

Syntax: ParseMLoad [BmMLoad, Prefix, Scale]

Where:
BmMLoad = Moment load data definition in Matrix format
Prefix = Moment load label prefix
Scale = Graphics plot scale

Scale = Graphics plot scale

Moment Load Plot Scale:

mScale := 2.5

Parse Moment Load Data for Graphics Output:

plot_mLoad := ParseMLoad(*mLoad*, "M", *mScale*)

5.) Parse 'Uniform Load Data' into Graphics Matrix Format:

Syntax: ParseQLoad [BmQLoad, Prefix, Scale]

Where:
BmQLoad = Uniform load data definition in Matrix format
Prefix = Uniform load label prefix
Scale = Graphics plot scale

Uniform Load Plot Scale:

uScale := 2.5

Parse Uniform Load Data for Graphics Output:

plot_uLoad := ParseQLoad(*uLoad*, "w", *uScale*)

5.) Parse 'Trapezoidal Load Data' into Graphics Matrix Format:

Syntax: ParseTLoad = [BmTLoad, Prefix, Scale]

Where:
BmTLoad = Trapezoidal load data definition in Matrix format
Prefix = Trapezoidal load label prefix
Scale = Graphics plot scale

Trapezoidal Load Plot Scale:

tScale := 2.5

Parse Trapezoidal Load Data for Graphics Output:

plot_tLoad := ParseTLoad(*tLoad*, "t", *tScale*)

e.) PLOT BEAM, SUPPORTS, & LOADINGS:

Once DATA PARSING is done -- BEAM, together with SUPPORTS and LOADINGS, can now be plotted onto SMath Studio 2D chart using 'PlotBeam' command.

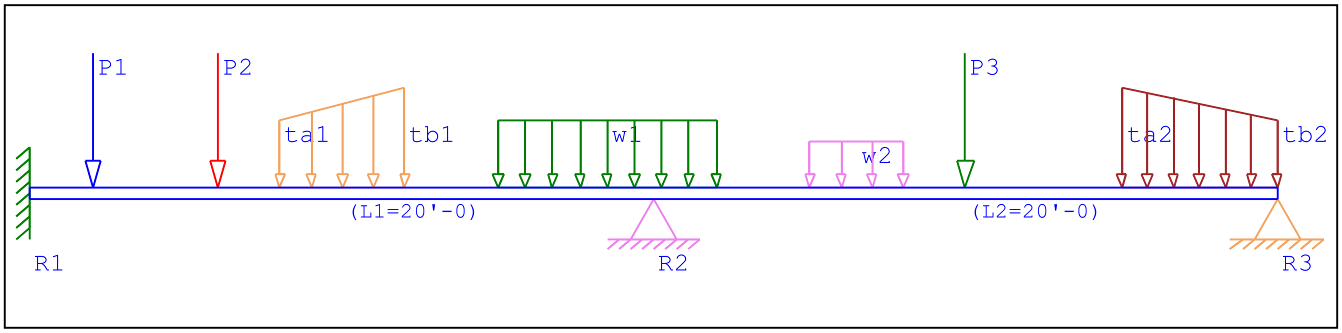
The command, specially developed by Redem Legaspi Jr for Crystal Steel, takes an infinite number of arguments -- with syntax shown below:

Syntax: PlotBeam(Arg1, Arg2, Arg3... ArgN)

Where:

Arg1, Arg2, Arg3... ArgN are arguments that correspond to 'Parsed Beam Data', 'Parsed Support Data', 'Parsed Concentrated Load Data', 'Parsed Uniform Load Data', 'Parsed Trapezodial Load Data', 'Parsed Moment Load Data', and so on. Arguments, however, can be ARRANGED IN ANY ORDER...

If supplied with correct parsed data, the'PlotBeam' command yields a realistic graphic representation of beam -- complete with supports and loadings as shown. From an engineering standpoint, we call it BEAM DIAGRAM...



PlotBeam(*plot_beam*, *plot_support*, *plot_uLoad*, *plot_pLoad*, *plot_tLoad*)

Invalid Support DETECTION (a fool-proof feature):

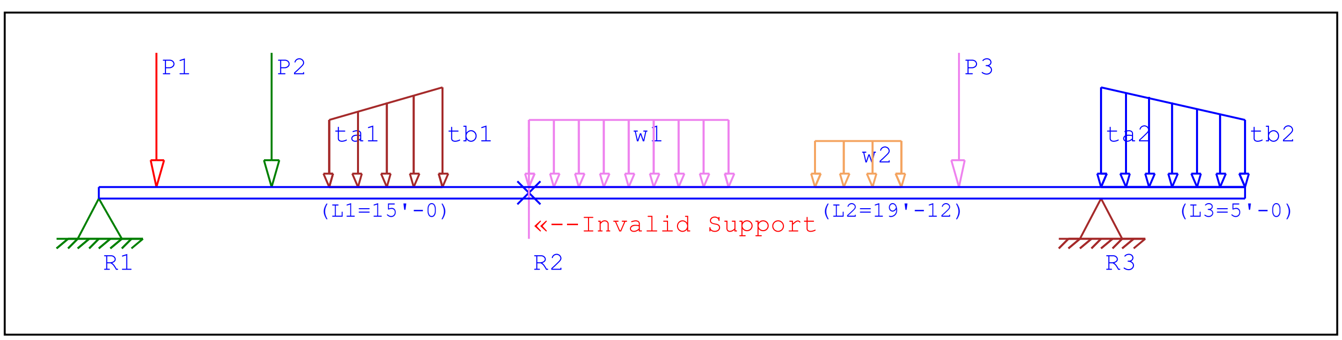
'PlotBeam' command prompts a 'graphics error warning' if an invalid support definition is encountered. An example is as shown below:

Invalid Support Data:

$$InvalidSupports := \begin{bmatrix} \text{"PIN"} & \text{"FIXED"} & \text{"PIN"} \\ 0\ ft & 15\ ft & 35\ ft \end{bmatrix}$$

Parse Invalid Support Data:

$$plotInvalidSupport := \text{ParseSupport}\left(InvalidSupports, L_b, \text{"R"}, bt, sScale\right)$$



PlotBeam(*plot_beam*, *plotInvalidSupport*, *plot_uLoad*, *plot_pLoad*, *plot_tLoad*)

The reason why support 'R2' is detected as invalid -- is that, by simple logic, FIXED support cannot be assigned anywhere along the beam span. It maybe assigned only to either end (or both ends) of the beam.

Dectecting this sort of engineering lapses early on while doing beam analysis

Detecting this sort of engineering lapses early on while doing beam analysis may help ease an engineer's burden and makes ones lovelife more colorful and more enjoyable. (a non sequitur?? he he he!!!)

There are lots of other 'fool-proof' features embedded in this application which may take a lot of space and time if completely discussed here. You will eventually discover and encounter those features as soon as you start using the program.

f.) PERFORM FINITE ELEMENT ANALYSIS:

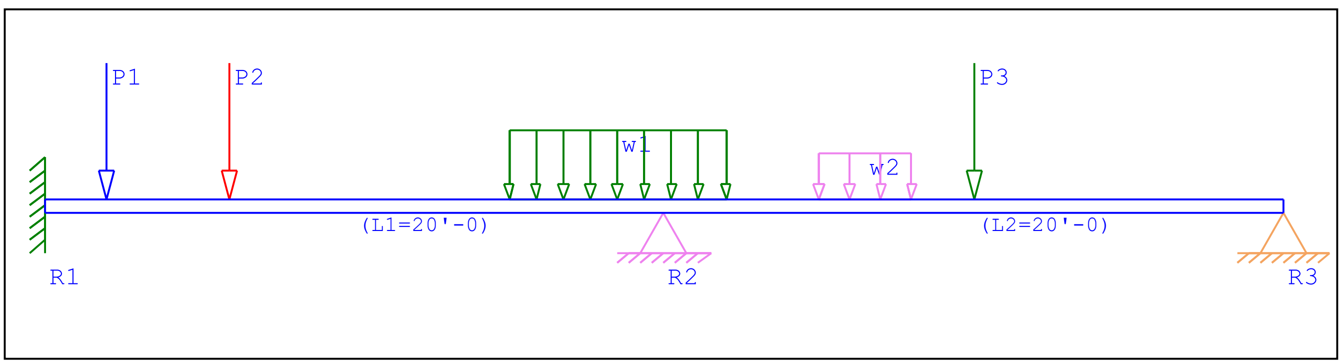
NOTE:
In this example I choose to apply 'CONCENTRATED LOAD' and 'UNIFORM LOAD' only due to my obvious preference for SIMPLICITY. (Simple is beautiful)

One has the complete discretion over what loads he wanted to appear on the BEAM DIAGRAM and what loads he wanted to be included in the analysis. This is done, for example, by specifying only the 'CONCENTRATED LOAD' and 'UNIFORM LOAD' onto 'PlotBeam' and 'AnalyzeBeamXXXX' commands as shown below.

This command call, for example,

PlotBeam(plot_beam, plot_support, plot_uLoad, plot_pLoad)

plots beam and supports with CONCENTRATED and UNIFORM loads only.



PlotBeam(*plot_beam*, *plot_support*, *plot_uLoad*, *plot_pLoad*)

FEM Analysis of beam is performed using either of the two commands:

1.) AnalyzeBeamCaprani

This command bears the name of a well-known structural engineer, Dr. Colin Caprani -- so named to give him due credit for writing the original source code of the FEABeam (Finite Element Analysis) computer program from which this application is adapted from. The program source code, fortunately, was written in Visual Basic -- my favorite programming language.

For those interested in programming, please visit http://www.colincaprani.com/.

With the code upgraded to VB.NET and implemented in SMath Studio platform, the program still bears about 50 percent of the original code attributed the program still bears about 50 percent of the original code attributed to Dr. Colin Caprani. The other 50 percent comprised the implementation program code that belongs to Redem Legaspi Jr.

The bottomline here is -- I just wanna say thanks to Dr. Caprani.

Limitations:
The program has a known bug that I just recently discovered. This bug has already been brought to the attention of Dr. Colin Caprani, but up to this writing no reply has been received yet from the kind engineer.

Here is the bug: with TRAPEZOIDAL LOADING applied, the program returns wrong results for both shear and moment including the deflection. I have made several attempts modifying the code to correct the problem but with no apparent success.

While this problem on trapezoidal loading is still being resolved, please refrain from applying trapezoidal load when doing beam analysis.

The command adapts the following syntax:

Syntax: AnalyzeBeamCaprani(Span, sData, pData, mData, tData, uData, eData, iData)

Where:
Span = Span of Beam
sData = Support Data in Matrix Format
pData = Concentrated Load Data in Matrix Format
mData = Moment Load Data in Matrix Format
tData = Trapezoidal Load Data in Matrix Format
uData = Uniform Load Data in Matrix Format
eData = Modulus of Elasticity Data in matrix Format
iData = Moment of Inertia Data in Matrix Format

If you opt not to apply any of the loadings, just supply the argument field with a pair of QUOTES ("") as shown below.

Perform FINITE ELEMENT ANALYSIS based on Dr. Colin Caprani's work:

$AnL_{caprani} := \text{AnalyzeBeamCaprani}\left(L_b, Supports, pLoad, "", "", uLoad, EData, IData\right)$

2.) AnalyzeBeamYakpol

Just like the 'AnalyzeBeamCaprani' command, this command is so named to give due credit to Yakov Polyakov -- a generous russian engineer and a very good programmer. This command has no problem dealing with trapezoidal loads.

The command adapts the following syntax:

Syntax: AnalyzeBeamYakpol(Span, sData, pData, mData, tData, uData, eData, iData)

Where:
Span = Span of Beam
sData = Support Data in Matrix Format
pData = Concentrated Load Data in Matrix Format

pData = Concentrated Load Data in Matrix Format
mData = Moment Load Data in Matrix Format
tData = Trapezoidal Load Data in Matrix Format
uData = Uniform Load Data in Matrix Format
eData = Modulus of Elasticity Data in SINGLE FLOATING VALUE
iData = Moment of Inertia Data in SINGLE FLOATING VALUE

To emphasize, the only difference that 'AnalyzeBeamCaprani' and 'AnalyzeBeamYakpol' commands have -- is the 'eData' and 'iData' inputs. In AnalyzeBeamCaprani command, the eData and iData inputs are both in MATRIX FORMAT.

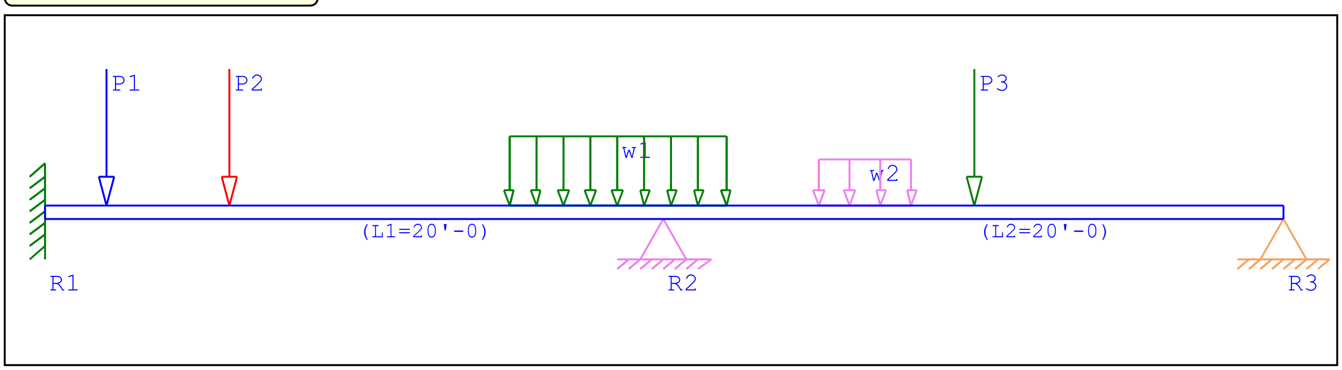
In AnalyzeBeamYakpol command, on the other hand, the eData and iData inputs are both in single FLOATING POINTS VALUES. In other words, the command does NOT take VARYING MODULUS OF ELASTICITY and VARYING MOMENT OF INERTIA throughout the beam.

If you opt not to apply any of the loadings, just supply the argument field with a pair of QUOTES ("") as shown below.

$$AnL_{yakpol} := \text{AnalyzeBeamYakpol}\left(L_b, Supports, pLoad, \text{""}, \text{""}, uLoad, 29000\ \text{ksi}, 23300\ \text{in}^4\right)$$

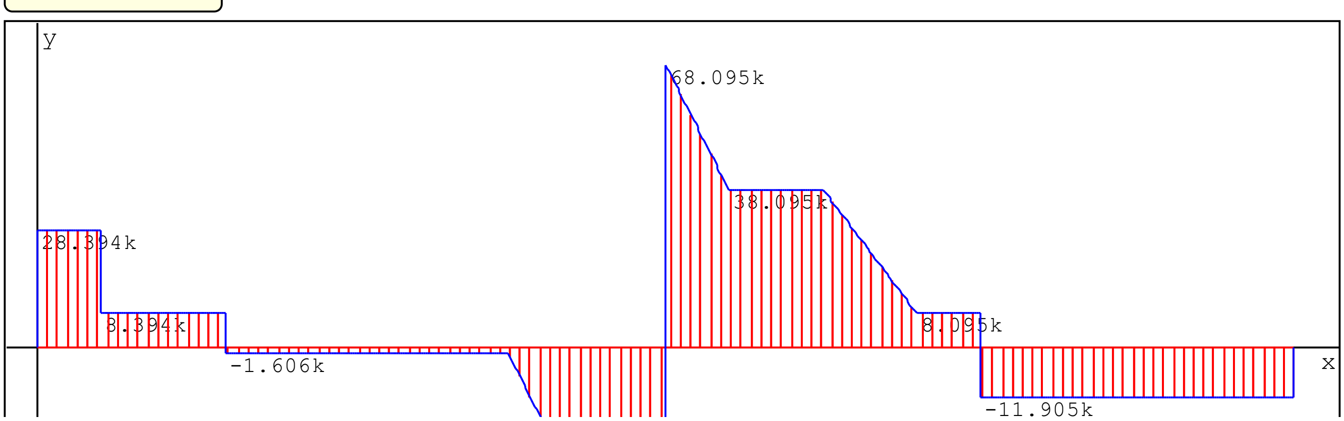
g.) SHEAR & MOMENT DIAGRAMS AND DEFLECTION:

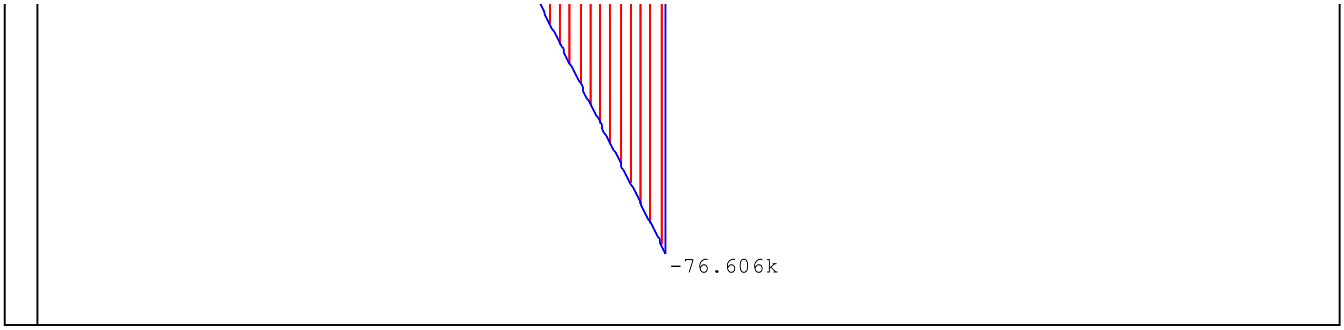
BEAM LOADING DIAGRAM:



PlotBeam(*plot_beam*, *plot_support*, *plot_uLoad*, *plot_pLoad*)

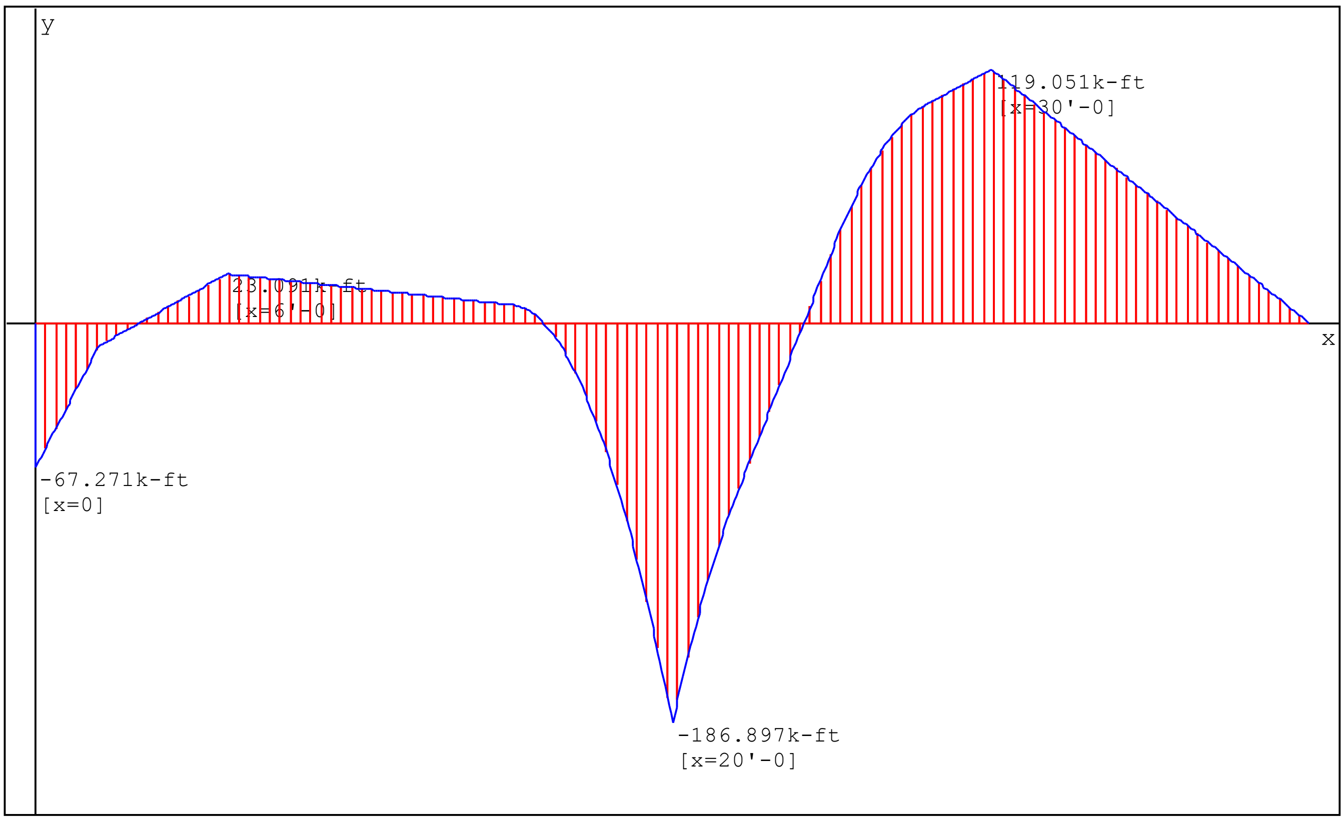
SHEAR DIAGRAM:





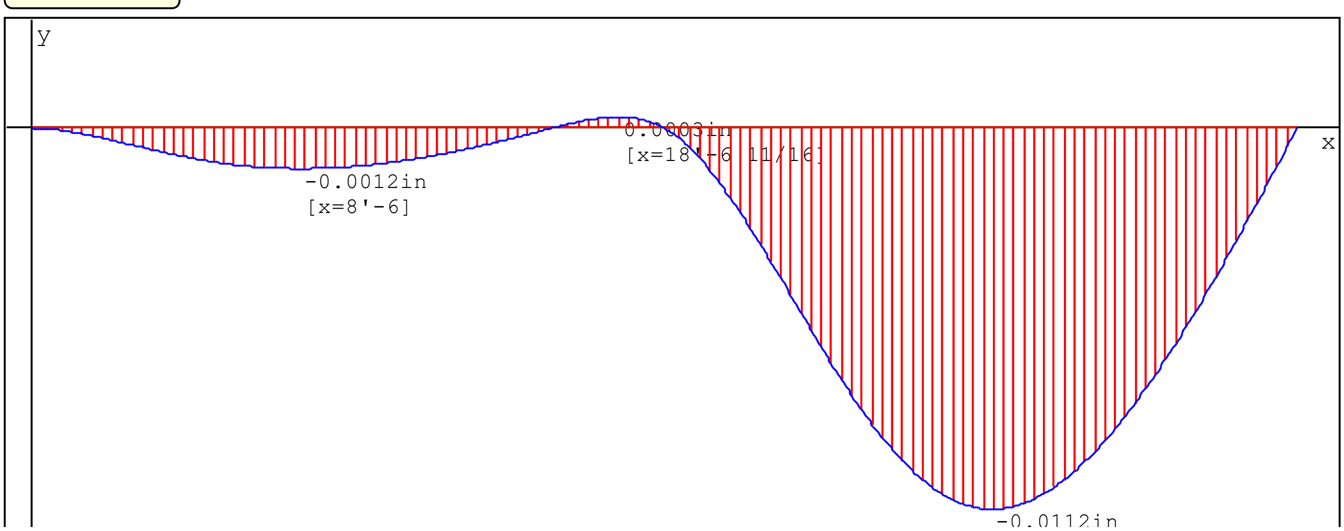
PlotShearDiagram $\left(AnL_{caprani}\right)$

MOMENT DIAGRAM:



PlotMomentDiagram $\left(AnL_{caprani}\right)$

DEFLECTION:



[x=30'-4]

PlotDeflection$\left(AnL_{caprani}\right)$

h.) SUPPORT REACTIONS:

Support Reactions may be determined using the following commands:

Reaction at Support R1

$R_1 :=$ SupportReaction$\left(AnL_{caprani}, 0\ ft\right)$

$R_1 = 28.394\ kip$

Reaaction at Support R2

$R_2 :=$ SupportReaction$\left(AnL_{caprani}, 20\ ft\right)$

$R_2 = 144.701\ kip$

Reaction at Support R3

$R_3 :=$ SupportReaction$\left(AnL_{caprani}, 40\ ft\right)$

$R_3 = 11.905\ kip$

Moment at Support R1

$M_1 :=$ SupportMoment$\left(AnL_{caprani}, 0\ ft\right)$

$M_1 = -67.271\ kip\ ft$

Moment at Support R2

$M_2 :=$ SupportMoment$\left(AnL_{caprani}, 20\ ft\right)$

$M_2 = -56.9662\ m\ kip$

Moment at Support R3

$M_3 :=$ SupportMoment$\left(AnL_{caprani}, 40\ ft\right)$

$M_3 = 0\ kip\ ft$

Where the first argument, AnLcaprani, is the variable that contains the analysis result matrix returned by either 'AnalyzeBeamCaprani' or 'AnalyzeBeamYakpol' command.

The second argument is the distance along the beam that corresponds to the location of support.

h.) QUERIES:

Shear, Bending Moment and Deflection along the beam span can be determined using the following commands:

Shear at any point along the beam:

$\text{GetShear}\left(AnL_{caprani}, 6\ \text{ft}\right) = 8.394\ \text{kip}$

Bending Moment at any point along the beam:

$\text{GetBendingMoment}\left(AnL_{caprani}, 6.0\ \text{ft}\right) = 23.091\ \text{kip ft}$

Deflection at any point along the beam:

$\text{GetDeflection}\left(AnL_{caprani}, 30.5\ \text{ft}\right) = -0.011221\ \text{in}$

Where the first argument, AnLcaprani,is the analysis result matrix returned by either 'AnalyzeBeamCaprani' or 'AnalyzeBeamYakpol' command.

The second argument is the distance along the beam where Shear, Bending Moment, or Deflection is to be determined.

i.) MAXIMUMS:

To determine the maximum values, use the following commands:

Maximum Bending Moment:

$M_{max} := \text{MaxBendingMoment}\left(AnL_{caprani}\right)$

$M_{max} = -186.897\ \text{kip ft}$

Maximum Positive Bending Moment:

$M_{maxpositive} := \text{MaxPositiveBendingMoment}\left(AnL_{caprani}\right)$

$M_{maxpositive} = 119.051\ \text{kip ft}$

Maximum Negative Bending Moment:

$M_{maxnegative} := \text{MaxNegativeBendingMoment}\left(AnL_{caprani}\right)$

$M_{maxnegative} = -186.897\ \text{kip ft}$

Maximum Shear:

$V_{max} := \text{MaxShear}\left(AnL_{caprani}\right)$

$V_{max} = -76.606\ \text{kip}$

Maximum Positive Shear:

$V_{maxpositive} := \text{MaxPositiveShear}\left(AnL_{caprani}\right)$

$V_{maxpositive} = 68.095$ kip

Maximum Negative Shear:

$V_{maxnegative} := \text{MaxNegativeShear}\left(AnL_{caprani}\right)$

$V_{maxnegative} = -76.606$ kip

Maximum Deflection:

$D_{max} := \text{MaxDeflection}\left(AnL_{caprani}\right)$

$D_{max} = -0.01122466$ in

Maximum Positive Deflection:

$D_{maxpositive} := \text{MaxPositiveDeflection}\left(AnL_{caprani}\right)$

$D_{maxpositive} = 0.00032872$ in

Maximum Negative Deflection:

$D_{maxnegative} := \text{MaxNegativeDeflection}\left(AnL_{caprani}\right)$

$D_{maxnegative} = -0.01122466$ in

j.) LOCATIONS OF MAXIMUMS:

Use the following commands determine the locations of the MAXIMUMS along the the beam span:

Maximum Bending Moment Location:

$LM_{max} := \text{MaxBendingMomentLocation}\left(AnL_{caprani}\right)$

$LM_{max} = 20$ ft

Maximum Positive Bending Moment Location:

$LM_{maxpositive} := \text{MaxPositiveBendingMomentLocation}\left(AnL_{caprani}\right)$

$LM_{maxpositive} = 30$ ft

Maximum Negative Bending Moment Location:

$LM_{maxnegative} := \text{MaxNegativeBendingMomentLocation}\left(AnL_{caprani}\right)$

$LM_{maxnegative} = 20 \text{ ft}$

Maximum Deflection Location:

$LD_{max} := \text{MaxDeflectionLocation}\left(AnL_{caprani}\right)$

$LD_{max} = 30.333 \text{ ft}$

Maximum Positive Deflection Location:

$LD_{maxpositive} := \text{MaxPositiveDeflectionLocation}\left(AnL_{caprani}\right)$

$LD_{maxpositive} = 18.556 \text{ ft}$

Maximum Negative Deflection Location:

$LD_{maxnegative} := \text{MaxNegativeDeflectionLocation}\left(AnL_{caprani}\right)$

$LD_{maxnegative} = 30.333 \text{ ft}$

k.) FEM RESULTS DATA STRUCTURE:

WANT TO KNOW WHAT IS INSIDE THE VARIABLE AnLcaprani? And how the ANALYSIS RESULTS in matrix format are being STRUCTURED AND STORED in a variable?

WELL, FIND IT BELOW...

Please note that retrieving values from these results is far more complex than returning 'these results' itself.

One of those complex things done in this program is the 'routine' that creates diagrams (shear, moment & deflection) & intelligently label values by itself.

There are an endless list of possible upgrades to this program, which, as of the moment are beyond the author's grasp. Your comments and suggestions,therefore, are paramount to make these upgrades possible...

$$\left(\begin{bmatrix} 0 & 28.394 \\ 20 & 144.701 \\ 40 & 11.905 \end{bmatrix} \quad \begin{bmatrix} 0 & -67.271 \end{bmatrix} \right.$$

0	67.271
20	−186.897
40	0

0	0
0	0
0	28.394
0.056	28.394
0.056	28.394
0.111	28.394
0.111	28.394
0.167	28.394
0.167	28.394
0.222	28.394
0.222	28.394
0.278	28.394
0.278	28.394
0.333	28.394
0.333	28.394
0.389	28.394
0.389	28.394
0.444	28.394
0.444	28.394
0.5	28.394
0.5	28.394
0.556	28.394
0.556	28.394
0.611	28.394
0.611	28.394
0.667	28.394
0.667	28.394
0.722	28.394
0.722	28.394
0.778	28.394
0.778	28.394
0.833	28.394
0.833	28.394
0.889	28.394
0.889	28.394
0.944	28.394
0.944	28.394
1	28.394
1	28.394
1.056	28.394
1.056	28.394
1.111	28.394
1.111	28.394
1.167	28.394
1.167	28.394
1.222	28.394
1.222	28.394
1.278	28.394
1.278	28.394
1.333	28.394
1.333	28.394
1.389	28.394
1.389	28.394
1.444	28.394

1.444	28.394
1.5	28.394
1.5	28.394
1.556	28.394
1.556	28.394
1.611	28.394
1.611	28.394
1.667	28.394
1.667	28.394
1.722	28.394
1.722	28.394
1.778	28.394
1.778	28.394
1.833	28.394
1.833	28.394
1.889	28.394
1.889	28.394
1.944	28.394
1.944	28.394
2	28.394
2	8.394
2.056	8.394
2.056	8.394
2.111	8.394
2.111	8.394
2.167	8.394
2.167	8.394
2.222	8.394
2.222	8.394
2.278	8.394
2.278	8.394
2.333	8.394
2.333	8.394
2.389	8.394
2.389	8.394
2.444	8.394
2.444	8.394
2.5	8.394
2.5	8.394
2.556	8.394
2.556	8.394
2.611	8.394
2.611	8.394
2.667	8.394
2.667	8.394
2.722	8.394
2.722	8.394
2.778	8.394
2.778	8.394
2.833	8.394
2.833	8.394
2.889	8.394
2.889	8.394
2.944	8.394
2.944	8.394
3	8.394

3	8.394
3.056	8.394
3.056	8.394
3.111	8.394
3.111	8.394
3.167	8.394
3.167	8.394
3.222	8.394
3.222	8.394
3.278	8.394
3.278	8.394
3.333	8.394
3.333	8.394
3.389	8.394
3.389	8.394
3.444	8.394
3.444	8.394
3.5	8.394
3.5	8.394
3.556	8.394
3.556	8.394
3.611	8.394
3.611	8.394
3.667	8.394
3.667	8.394
3.722	8.394
3.722	8.394
3.778	8.394
3.778	8.394
3.833	8.394
3.833	8.394
3.889	8.394
3.889	8.394
3.944	8.394
3.944	8.394
4	8.394
4	8.394
4.056	8.394
4.056	8.394
4.111	8.394
4.111	8.394
4.167	8.394
4.167	8.394
4.222	8.394
4.222	8.394
4.278	8.394
4.278	8.394
4.333	8.394
4.333	8.394
4.389	8.394
4.389	8.394
4.444	8.394
4.444	8.394
4.5	8.394
4.5	8.394
4.556	8.394
4.556	8.394

4.611	8.394
4.611	8.394
4.667	8.394
4.667	8.394
4.722	8.394
4.722	8.394
4.778	8.394
4.778	8.394
4.833	8.394
4.833	8.394
4.889	8.394
4.889	8.394
4.944	8.394
4.944	8.394
5	8.394
5	8.394
5.056	8.394
5.056	8.394
5.111	8.394
5.111	8.394
5.167	8.394
5.167	8.394
5.222	8.394
5.222	8.394
5.278	8.394
5.278	8.394
5.333	8.394
5.333	8.394
5.389	8.394
5.389	8.394
5.444	8.394
5.444	8.394
5.5	8.394
5.5	8.394
5.556	8.394
5.556	8.394
5.611	8.394
5.611	8.394
5.667	8.394
5.667	8.394
5.722	8.394
5.722	8.394
5.778	8.394
5.778	8.394
5.833	8.394
5.833	8.394
5.889	8.394
5.889	8.394
5.944	8.394
5.944	8.394
6	8.394
6	−1.606
6.056	−1.606
6.056	−1.606
6.111	−1.606
6.111	−1.606

6.167	−1.606
6.167	−1.606
6.222	−1.606
6.222	−1.606
6.278	−1.606
6.278	−1.606
6.333	−1.606
6.333	−1.606
6.389	−1.606
6.389	−1.606
6.444	−1.606
6.444	−1.606
6.5	−1.606
6.5	−1.606
6.556	−1.606
6.556	−1.606
6.611	−1.606
6.611	−1.606
6.667	−1.606
6.667	−1.606
6.722	−1.606
6.722	−1.606
6.778	−1.606
6.778	−1.606
6.833	−1.606
6.833	−1.606
6.889	−1.606
6.889	−1.606
6.944	−1.606
6.944	−1.606
7	−1.606
7	−1.606
7.056	−1.606
7.056	−1.606
7.111	−1.606
7.111	−1.606
7.167	−1.606
7.167	−1.606
7.222	−1.606
7.222	−1.606
7.278	−1.606
7.278	−1.606
7.333	−1.606
7.333	−1.606
7.389	−1.606
7.389	−1.606
7.444	−1.606
7.444	−1.606
7.5	−1.606
7.5	−1.606
7.556	−1.606
7.556	−1.606
7.611	−1.606
7.611	−1.606
7.667	−1.606
7.667	−1.606
7.722	−1.606

7.722	−1.606
7.722	−1.606
7.778	−1.606
7.778	−1.606
7.833	−1.606
7.833	−1.606
7.889	−1.606
7.889	−1.606
7.944	−1.606
7.944	−1.606
8	−1.606
8	−1.606
8.056	−1.606
8.056	−1.606
8.111	−1.606
8.111	−1.606
8.167	−1.606
8.167	−1.606
8.222	−1.606
8.222	−1.606
8.278	−1.606
8.278	−1.606
8.333	−1.606
8.333	−1.606
8.389	−1.606
8.389	−1.606
8.444	−1.606
8.444	−1.606
8.5	−1.606
8.5	−1.606
8.556	−1.606
8.556	−1.606
8.611	−1.606
8.611	−1.606
8.667	−1.606
8.667	−1.606
8.722	−1.606
8.722	−1.606
8.778	−1.606
8.778	−1.606
8.833	−1.606
8.833	−1.606
8.889	−1.606
8.889	−1.606
8.944	−1.606
8.944	−1.606
9	−1.606
9	−1.606
9.056	−1.606
9.056	−1.606
9.111	−1.606
9.111	−1.606
9.167	−1.606
9.167	−1.606
9.222	−1.606
9.222	−1.606
9.278	−1.606

9.278	−1.606
9.333	−1.606
9.333	−1.606
9.389	−1.606
9.389	−1.606
9.444	−1.606
9.444	−1.606
9.5	−1.606
9.5	−1.606
9.556	−1.606
9.556	−1.606
9.611	−1.606
9.611	−1.606
9.667	−1.606
9.667	−1.606
9.722	−1.606
9.722	−1.606
9.778	−1.606
9.778	−1.606
9.833	−1.606
9.833	−1.606
9.889	−1.606
9.889	−1.606
9.944	−1.606
9.944	−1.606
10	−1.606
10	−1.606
10.056	−1.606
10.056	−1.606
10.111	−1.606
10.111	−1.606
10.167	−1.606
10.167	−1.606
10.222	−1.606
10.222	−1.606
10.278	−1.606
10.278	−1.606
10.333	−1.606
10.333	−1.606
10.389	−1.606
10.389	−1.606
10.444	−1.606
10.444	−1.606
10.5	−1.606
10.5	−1.606
10.556	−1.606
10.556	−1.606
10.611	−1.606
10.611	−1.606
10.667	−1.606
10.667	−1.606
10.722	−1.606
10.722	−1.606
10.778	−1.606
10.778	−1.606
10.833	−1.606
10.833	−1.606

10.833	−1.606
10.889	−1.606
10.889	−1.606
10.944	−1.606
10.944	−1.606
11	−1.606
11	−1.606
11.056	−1.606
11.056	−1.606
11.111	−1.606
11.111	−1.606
11.167	−1.606
11.167	−1.606
11.222	−1.606
11.222	−1.606
11.278	−1.606
11.278	−1.606
11.333	−1.606
11.333	−1.606
11.389	−1.606
11.389	−1.606
11.444	−1.606
11.444	−1.606
11.5	−1.606
11.5	−1.606
11.556	−1.606
11.556	−1.606
11.611	−1.606
11.611	−1.606
11.667	−1.606
11.667	−1.606
11.722	−1.606
11.722	−1.606
11.778	−1.606
11.778	−1.606
11.833	−1.606
11.833	−1.606
11.889	−1.606
11.889	−1.606
11.944	−1.606
11.944	−1.606
12	−1.606
12	−1.606
12.056	−1.606
12.056	−1.606
12.111	−1.606
12.111	−1.606
12.167	−1.606
12.167	−1.606
12.222	−1.606
12.222	−1.606
12.278	−1.606
12.278	−1.606
12.333	−1.606
12.333	−1.606
12.389	−1.606
12.389	−1.606

12.444	−1.606
12.444	−1.606
12.5	−1.606
12.5	−1.606
12.556	−1.606
12.556	−1.606
12.611	−1.606
12.611	−1.606
12.667	−1.606
12.667	−1.606
12.722	−1.606
12.722	−1.606
12.778	−1.606
12.778	−1.606
12.833	−1.606
12.833	−1.606
12.889	−1.606
12.889	−1.606
12.944	−1.606
12.944	−1.606
13	−1.606
13	−1.606
13.056	−1.606
13.056	−1.606
13.111	−1.606
13.111	−1.606
13.167	−1.606
13.167	−1.606
13.222	−1.606
13.222	−1.606
13.278	−1.606
13.278	−1.606
13.333	−1.606
13.333	−1.606
13.389	−1.606
13.389	−1.606
13.444	−1.606
13.444	−1.606
13.5	−1.606
13.5	−1.606
13.556	−1.606
13.556	−1.606
13.611	−1.606
13.611	−1.606
13.667	−1.606
13.667	−1.606
13.722	−1.606
13.722	−1.606
13.778	−1.606
13.778	−1.606
13.833	−1.606
13.833	−1.606
13.889	−1.606
13.889	−1.606
13.944	−1.606
13.944	−1.606
14	−1.606

14	−1.606
14	−1.606
14.056	−1.606
14.056	−1.606
14.111	−1.606
14.111	−1.606
14.167	−1.606
14.167	−1.606
14.222	−1.606
14.222	−1.606
14.278	−1.606
14.278	−1.606
14.333	−1.606
14.333	−1.606
14.389	−1.606
14.389	−1.606
14.444	−1.606
14.444	−1.606
14.5	−1.606
14.5	−1.606
14.556	−1.606
14.556	−1.606
14.611	−1.606
14.611	−1.606
14.667	−1.606
14.667	−1.606
14.722	−1.606
14.722	−1.606
14.778	−1.606
14.778	−1.606
14.833	−1.606
14.833	−1.606
14.889	−1.606
14.889	−1.606
14.944	−1.606
14.944	−1.606
15	−1.606
15	−1.606
15.056	−2.44
15.056	−2.44
15.111	−3.273
15.111	−3.273
15.167	−4.106
15.167	−4.106
15.222	−4.94
15.222	−4.94
15.278	−5.773
15.278	−5.773
15.333	−6.606
15.333	−6.606
15.389	−7.44
15.389	−7.44
15.444	−8.273
15.444	−8.273
15.5	−9.106
15.5	−9.106
15.556	−9.94

15.556	−9.94
15.611	−10.773
15.611	−10.773
15.667	−11.606
15.667	−11.606
15.722	−12.44
15.722	−12.44
15.778	−13.273
15.778	−13.273
15.833	−14.106
15.833	−14.106
15.889	−14.94
15.889	−14.94
15.944	−15.773
15.944	−15.773
16	−16.606
16	−16.606
16.056	−17.44
16.056	−17.44
16.111	−18.273
16.111	−18.273
16.167	−19.106
16.167	−19.106
16.222	−19.94
16.222	−19.94
16.278	−20.773
16.278	−20.773
16.333	−21.606
16.333	−21.606
16.389	−22.44
16.389	−22.44
16.444	−23.273
16.444	−23.273
16.5	−24.106
16.5	−24.106
16.556	−24.94
16.556	−24.94
16.611	−25.773
16.611	−25.773
16.667	−26.606
16.667	−26.606
16.722	−27.44
16.722	−27.44
16.778	−28.273
16.778	−28.273
16.833	−29.106
16.833	−29.106
16.889	−29.94
16.889	−29.94
16.944	−30.773
16.944	−30.773
17	−31.606
17	−31.606
17.056	−32.44
17.056	−32.44
17.111	−33.273

17.111	−33.273
17.167	−34.106
17.167	−34.106
17.222	−34.94
17.222	−34.94
17.278	−35.773
17.278	−35.773
17.333	−36.606
17.333	−36.606
17.389	−37.44
17.389	−37.44
17.444	−38.273
17.444	−38.273
17.5	−39.106
17.5	−39.106
17.556	−39.94
17.556	−39.94
17.611	−40.773
17.611	−40.773
17.667	−41.606
17.667	−41.606
17.722	−42.44
17.722	−42.44
17.778	−43.273
17.778	−43.273
17.833	−44.106
17.833	−44.106
17.889	−44.94
17.889	−44.94
17.944	−45.773
17.944	−45.773
18	−46.606
18	−46.606
18.056	−47.44
18.056	−47.44
18.111	−48.273
18.111	−48.273
18.167	−49.106
18.167	−49.106
18.222	−49.94
18.222	−49.94
18.278	−50.773
18.278	−50.773
18.333	−51.606
18.333	−51.606
18.389	−52.44
18.389	−52.44
18.444	−53.273
18.444	−53.273
18.5	−54.106
18.5	−54.106
18.556	−54.94
18.556	−54.94
18.611	−55.773
18.611	−55.773
18.667	−56.606
18.667	−56.606

18.722	−57.44
18.722	−57.44
18.778	−58.273
18.778	−58.273
18.833	−59.106
18.833	−59.106
18.889	−59.94
18.889	−59.94
18.944	−60.773
18.944	−60.773
19	−61.606
19	−61.606
19.056	−62.44
19.056	−62.44
19.111	−63.273
19.111	−63.273
19.167	−64.106
19.167	−64.106
19.222	−64.94
19.222	−64.94
19.278	−65.773
19.278	−65.773
19.333	−66.606
19.333	−66.606
19.389	−67.44
19.389	−67.44
19.444	−68.273
19.444	−68.273
19.5	−69.106
19.5	−69.106
19.556	−69.94
19.556	−69.94
19.611	−70.773
19.611	−70.773
19.667	−71.606
19.667	−71.606
19.722	−72.44
19.722	−72.44
19.778	−73.273
19.778	−73.273
19.833	−74.106
19.833	−74.106
19.889	−74.94
19.889	−74.94
19.944	−75.773
19.944	−75.773
20	−76.606
20	68.095
20.056	67.262
20.056	67.262
20.111	66.428
20.111	66.428
20.167	65.595
20.167	65.595
20.222	64.762
20.222	64.762

20.278	63.928
20.278	63.928
20.333	63.095
20.333	63.095
20.389	62.262
20.389	62.262
20.444	61.428
20.444	61.428
20.5	60.595
20.5	60.595
20.556	59.762
20.556	59.762
20.611	58.928
20.611	58.928
20.667	58.095
20.667	58.095
20.722	57.262
20.722	57.262
20.778	56.428
20.778	56.428
20.833	55.595
20.833	55.595
20.889	54.762
20.889	54.762
20.944	53.928
20.944	53.928
21	53.095
21	53.095
21.056	52.262
21.056	52.262
21.111	51.428
21.111	51.428
21.167	50.595
21.167	50.595
21.222	49.762
21.222	49.762
21.278	48.928
21.278	48.928
21.333	48.095
21.333	48.095
21.389	47.262
21.389	47.262
21.444	46.428
21.444	46.428
21.5	45.595
21.5	45.595
21.556	44.762
21.556	44.762
21.611	43.928
21.611	43.928
21.667	43.095
21.667	43.095
21.722	42.262
21.722	42.262
21.778	41.428
21.778	41.428
21.833	40.595

21.833	40.595
21.889	39.762
21.889	39.762
21.944	38.928
21.944	38.928
22	38.095
22	38.095
22.056	38.095
22.056	38.095
22.111	38.095
22.111	38.095
22.167	38.095
22.167	38.095
22.222	38.095
22.222	38.095
22.278	38.095
22.278	38.095
22.333	38.095
22.333	38.095
22.389	38.095
22.389	38.095
22.444	38.095
22.444	38.095
22.5	38.095
22.5	38.095
22.556	38.095
22.556	38.095
22.611	38.095
22.611	38.095
22.667	38.095
22.667	38.095
22.722	38.095
22.722	38.095
22.778	38.095
22.778	38.095
22.833	38.095
22.833	38.095
22.889	38.095
22.889	38.095
22.944	38.095
22.944	38.095
23	38.095
23	38.095
23.056	38.095
23.056	38.095
23.111	38.095
23.111	38.095
23.167	38.095
23.167	38.095
23.222	38.095
23.222	38.095
23.278	38.095
23.278	38.095
23.333	38.095
23.333	38.095
23.389	38.095

23.389	38.095
23.444	38.095
23.444	38.095
23.5	38.095
23.5	38.095
23.556	38.095
23.556	38.095
23.611	38.095
23.611	38.095
23.667	38.095
23.667	38.095
23.722	38.095
23.722	38.095
23.778	38.095
23.778	38.095
23.833	38.095
23.833	38.095
23.889	38.095
23.889	38.095
23.944	38.095
23.944	38.095
24	38.095
24	38.095
24.056	38.095
24.056	38.095
24.111	38.095
24.111	38.095
24.167	38.095
24.167	38.095
24.222	38.095
24.222	38.095
24.278	38.095
24.278	38.095
24.333	38.095
24.333	38.095
24.389	38.095
24.389	38.095
24.444	38.095
24.444	38.095
24.5	38.095
24.5	38.095
24.556	38.095
24.556	38.095
24.611	38.095
24.611	38.095
24.667	38.095
24.667	38.095
24.722	38.095
24.722	38.095
24.778	38.095
24.778	38.095
24.833	38.095
24.833	38.095
24.889	38.095
24.889	38.095
24.944	38.095
24.944	38.095

24.944	38.095
25	38.095
25	38.095
25.056	37.539
25.056	37.539
25.111	36.984
25.111	36.984
25.167	36.428
25.167	36.428
25.222	35.873
25.222	35.873
25.278	35.317
25.278	35.317
25.333	34.762
25.333	34.762
25.389	34.206
25.389	34.206
25.444	33.65
25.444	33.65
25.5	33.095
25.5	33.095
25.556	32.539
25.556	32.539
25.611	31.984
25.611	31.984
25.667	31.428
25.667	31.428
25.722	30.873
25.722	30.873
25.778	30.317
25.778	30.317
25.833	29.762
25.833	29.762
25.889	29.206
25.889	29.206
25.944	28.65
25.944	28.65
26	28.095
26	28.095
26.056	27.539
26.056	27.539
26.111	26.984
26.111	26.984
26.167	26.428
26.167	26.428
26.222	25.873
26.222	25.873
26.278	25.317
26.278	25.317
26.333	24.762
26.333	24.762
26.389	24.206
26.389	24.206
26.444	23.65
26.444	23.65
26.5	23.095
26.5	23.095

26.556	22.539
26.556	22.539
26.611	21.984
26.611	21.984
26.667	21.428
26.667	21.428
26.722	20.873
26.722	20.873
26.778	20.317
26.778	20.317
26.833	19.762
26.833	19.762
26.889	19.206
26.889	19.206
26.944	18.65
26.944	18.65
27	18.095
27	18.095
27.056	17.539
27.056	17.539
27.111	16.984
27.111	16.984
27.167	16.428
27.167	16.428
27.222	15.873
27.222	15.873
27.278	15.317
27.278	15.317
27.333	14.762
27.333	14.762
27.389	14.206
27.389	14.206
27.444	13.65
27.444	13.65
27.5	13.095
27.5	13.095
27.556	12.539
27.556	12.539
27.611	11.984
27.611	11.984
27.667	11.428
27.667	11.428
27.722	10.873
27.722	10.873
27.778	10.317
27.778	10.317
27.833	9.762
27.833	9.762
27.889	9.206
27.889	9.206
27.944	8.65
27.944	8.65
28	8.095
28	8.095
28.056	8.095
28.056	8.095
28.111	8.095

28.111	8.095
28.111	8.095
28.167	8.095
28.167	8.095
28.222	8.095
28.222	8.095
28.278	8.095
28.278	8.095
28.333	8.095
28.333	8.095
28.389	8.095
28.389	8.095
28.444	8.095
28.444	8.095
28.5	8.095
28.5	8.095
28.556	8.095
28.556	8.095
28.611	8.095
28.611	8.095
28.667	8.095
28.667	8.095
28.722	8.095
28.722	8.095
28.778	8.095
28.778	8.095
28.833	8.095
28.833	8.095
28.889	8.095
28.889	8.095
28.944	8.095
28.944	8.095
29	8.095
29	8.095
29.056	8.095
29.056	8.095
29.111	8.095
29.111	8.095
29.167	8.095
29.167	8.095
29.222	8.095
29.222	8.095
29.278	8.095
29.278	8.095
29.333	8.095
29.333	8.095
29.389	8.095
29.389	8.095
29.444	8.095
29.444	8.095
29.5	8.095
29.5	8.095
29.556	8.095
29.556	8.095
29.611	8.095
29.611	8.095
29.667	8.095

29.667	8.095
29.722	8.095
29.722	8.095
29.778	8.095
29.778	8.095
29.833	8.095
29.833	8.095
29.889	8.095
29.889	8.095
29.944	8.095
29.944	8.095
30	8.095
30	−11.905
30.056	−11.905
30.056	−11.905
30.111	−11.905
30.111	−11.905
30.167	−11.905
30.167	−11.905
30.222	−11.905
30.222	−11.905
30.278	−11.905
30.278	−11.905
30.333	−11.905
30.333	−11.905
30.389	−11.905
30.389	−11.905
30.444	−11.905
30.444	−11.905
30.5	−11.905
30.5	−11.905
30.556	−11.905
30.556	−11.905
30.611	−11.905
30.611	−11.905
30.667	−11.905
30.667	−11.905
30.722	−11.905
30.722	−11.905
30.778	−11.905
30.778	−11.905
30.833	−11.905
30.833	−11.905
30.889	−11.905
30.889	−11.905
30.944	−11.905
30.944	−11.905
31	−11.905
31	−11.905
31.056	−11.905
31.056	−11.905
31.111	−11.905
31.111	−11.905
31.167	−11.905
31.167	−11.905
31.222	−11.905
31.222	−11.905

31.222	−11.905
31.278	−11.905
31.278	−11.905
31.333	−11.905
31.333	−11.905
31.389	−11.905
31.389	−11.905
31.444	−11.905
31.444	−11.905
31.5	−11.905
31.5	−11.905
31.556	−11.905
31.556	−11.905
31.611	−11.905
31.611	−11.905
31.667	−11.905
31.667	−11.905
31.722	−11.905
31.722	−11.905
31.778	−11.905
31.778	−11.905
31.833	−11.905
31.833	−11.905
31.889	−11.905
31.889	−11.905
31.944	−11.905
31.944	−11.905
32	−11.905
32	−11.905
32.056	−11.905
32.056	−11.905
32.111	−11.905
32.111	−11.905
32.167	−11.905
32.167	−11.905
32.222	−11.905
32.222	−11.905
32.278	−11.905
32.278	−11.905
32.333	−11.905
32.333	−11.905
32.389	−11.905
32.389	−11.905
32.444	−11.905
32.444	−11.905
32.5	−11.905
32.5	−11.905
32.556	−11.905
32.556	−11.905
32.611	−11.905
32.611	−11.905
32.667	−11.905
32.667	−11.905
32.722	−11.905
32.722	−11.905
32.778	−11.905
32.778	−11.905

32.833	−11.905
32.833	−11.905
32.889	−11.905
32.889	−11.905
32.944	−11.905
32.944	−11.905
33	−11.905
33	−11.905
33.056	−11.905
33.056	−11.905
33.111	−11.905
33.111	−11.905
33.167	−11.905
33.167	−11.905
33.222	−11.905
33.222	−11.905
33.278	−11.905
33.278	−11.905
33.333	−11.905
33.333	−11.905
33.389	−11.905
33.389	−11.905
33.444	−11.905
33.444	−11.905
33.5	−11.905
33.5	−11.905
33.556	−11.905
33.556	−11.905
33.611	−11.905
33.611	−11.905
33.667	−11.905
33.667	−11.905
33.722	−11.905
33.722	−11.905
33.778	−11.905
33.778	−11.905
33.833	−11.905
33.833	−11.905
33.889	−11.905
33.889	−11.905
33.944	−11.905
33.944	−11.905
34	−11.905
34	−11.905
34.056	−11.905
34.056	−11.905
34.111	−11.905
34.111	−11.905
34.167	−11.905
34.167	−11.905
34.222	−11.905
34.222	−11.905
34.278	−11.905
34.278	−11.905
34.333	−11.905
34.333	−11.905

34.389	−11.905
34.389	−11.905
34.444	−11.905
34.444	−11.905
34.5	−11.905
34.5	−11.905
34.556	−11.905
34.556	−11.905
34.611	−11.905
34.611	−11.905
34.667	−11.905
34.667	−11.905
34.722	−11.905
34.722	−11.905
34.778	−11.905
34.778	−11.905
34.833	−11.905
34.833	−11.905
34.889	−11.905
34.889	−11.905
34.944	−11.905
34.944	−11.905
35	−11.905
35	−11.905
35.056	−11.905
35.056	−11.905
35.111	−11.905
35.111	−11.905
35.167	−11.905
35.167	−11.905
35.222	−11.905
35.222	−11.905
35.278	−11.905
35.278	−11.905
35.333	−11.905
35.333	−11.905
35.389	−11.905
35.389	−11.905
35.444	−11.905
35.444	−11.905
35.5	−11.905
35.5	−11.905
35.556	−11.905
35.556	−11.905
35.611	−11.905
35.611	−11.905
35.667	−11.905
35.667	−11.905
35.722	−11.905
35.722	−11.905
35.778	−11.905
35.778	−11.905
35.833	−11.905
35.833	−11.905
35.889	−11.905
35.889	−11.905
35.944	−11.905

35.944	−11.905
36	−11.905
36	−11.905
36.056	−11.905
36.056	−11.905
36.111	−11.905
36.111	−11.905
36.167	−11.905
36.167	−11.905
36.222	−11.905
36.222	−11.905
36.278	−11.905
36.278	−11.905
36.333	−11.905
36.333	−11.905
36.389	−11.905
36.389	−11.905
36.444	−11.905
36.444	−11.905
36.5	−11.905
36.5	−11.905
36.556	−11.905
36.556	−11.905
36.611	−11.905
36.611	−11.905
36.667	−11.905
36.667	−11.905
36.722	−11.905
36.722	−11.905
36.778	−11.905
36.778	−11.905
36.833	−11.905
36.833	−11.905
36.889	−11.905
36.889	−11.905
36.944	−11.905
36.944	−11.905
37	−11.905
37	−11.905
37.056	−11.905
37.056	−11.905
37.111	−11.905
37.111	−11.905
37.167	−11.905
37.167	−11.905
37.222	−11.905
37.222	−11.905
37.278	−11.905
37.278	−11.905
37.333	−11.905
37.333	−11.905
37.389	−11.905
37.389	−11.905
37.444	−11.905
37.444	−11.905
37.5	−11.905

37.5	−11.905
37.556	−11.905
37.556	−11.905
37.611	−11.905
37.611	−11.905
37.667	−11.905
37.667	−11.905
37.722	−11.905
37.722	−11.905
37.778	−11.905
37.778	−11.905
37.833	−11.905
37.833	−11.905
37.889	−11.905
37.889	−11.905
37.944	−11.905
37.944	−11.905
38	−11.905
38	−11.905
38.056	−11.905
38.056	−11.905
38.111	−11.905
38.111	−11.905
38.167	−11.905
38.167	−11.905
38.222	−11.905
38.222	−11.905
38.278	−11.905
38.278	−11.905
38.333	−11.905
38.333	−11.905
38.389	−11.905
38.389	−11.905
38.444	−11.905
38.444	−11.905
38.5	−11.905
38.5	−11.905
38.556	−11.905
38.556	−11.905
38.611	−11.905
38.611	−11.905
38.667	−11.905
38.667	−11.905
38.722	−11.905
38.722	−11.905
38.778	−11.905
38.778	−11.905
38.833	−11.905
38.833	−11.905
38.889	−11.905
38.889	−11.905
38.944	−11.905
38.944	−11.905
39	−11.905
39	−11.905
39.056	−11.905
39.056	−11.905

39.111	−11.905
39.111	−11.905
39.167	−11.905
39.167	−11.905
39.222	−11.905
39.222	−11.905
39.278	−11.905
39.278	−11.905
39.333	−11.905
39.333	−11.905
39.389	−11.905
39.389	−11.905
39.444	−11.905
39.444	−11.905
39.5	−11.905
39.5	−11.905
39.556	−11.905
39.556	−11.905
39.611	−11.905
39.611	−11.905
39.667	−11.905
39.667	−11.905
39.722	−11.905
39.722	−11.905
39.778	−11.905
39.778	−11.905
39.833	−11.905
39.833	−11.905
39.889	−11.905
39.889	−11.905
39.944	−11.905
39.944	−11.905
40	−11.905
40	0

$AnL_{caprani} =$

0	0
0	−67.271
0.056	−65.694
0.111	−64.116
0.167	−62.539
0.222	−60.961
0.278	−59.384
0.333	−57.806
0.389	−56.229
0.444	−54.652
0.5	−53.074
0.556	−51.497
0.611	−49.919
0.667	−48.342
0.722	−46.764
0.778	−45.187
0.833	−43.61
0.889	−42.032
0.944	−40.455
1	−38.877
1.056	−37.3
1.111	−35.722

1.167	−34.145
1.222	−32.568
1.278	−30.99
1.333	−29.413
1.389	−27.835
1.444	−26.258
1.5	−24.68
1.556	−23.103
1.611	−21.526
1.667	−19.948
1.722	−18.371
1.778	−16.793
1.833	−15.216
1.889	−13.638
1.944	−12.061
2	−10.484
2.056	−10.017
2.111	−9.551
2.167	−9.085
2.222	−8.618
2.278	−8.152
2.333	−7.686
2.389	−7.219
2.444	−6.753
2.5	−6.287
2.556	−5.82
2.611	−5.354
2.667	−4.888
2.722	−4.421
2.778	−3.955
2.833	−3.489
2.889	−3.023
2.944	−2.556
3	−2.09
3.056	−1.624
3.111	−1.157
3.167	−0.691
3.222	−0.225
3.278	0.242
3.333	0.708
3.389	1.174
3.444	1.641
3.5	2.107
3.556	2.573
3.611	3.04
3.667	3.506
3.722	3.972
3.778	4.439
3.833	4.905
3.889	5.371
3.944	5.837
4	6.304
4.056	6.77
4.111	7.236
4.167	7.703
4.222	8.169
[illegible]	[illegible]

4.278	8.635
4.333	9.102
4.389	9.568
4.444	10.034
4.5	10.501
4.556	10.967
4.611	11.433
4.667	11.9
4.722	12.366
4.778	12.832
4.833	13.299
4.889	13.765
4.944	14.231
5	14.697
5.056	15.164
5.111	15.63
5.167	16.096
5.222	16.563
5.278	17.029
5.333	17.495
5.389	17.962
5.444	18.428
5.5	18.894
5.556	19.361
5.611	19.827
5.667	20.293
5.722	20.76
5.778	21.226
5.833	21.692
5.889	22.159
5.944	22.625
6	23.091
6.056	23.002
6.111	22.913
6.167	22.823
6.222	22.734
6.278	22.645
6.333	22.556
6.389	22.466
6.444	22.377
6.5	22.288
6.556	22.199
6.611	22.11
6.667	22.02
6.722	21.931
6.778	21.842
6.833	21.753
6.889	21.663
6.944	21.574
7	21.485
7.056	21.396
7.111	21.306
7.167	21.217
7.222	21.128
7.278	21.039
7.333	20.949
7.389	20.86

7.444	20.771
7.5	20.682
7.556	20.592
7.611	20.503
7.667	20.414
7.722	20.325
7.778	20.235
7.833	20.146
7.889	20.057
7.944	19.968
8	19.879
8.056	19.789
8.111	19.7
8.167	19.611
8.222	19.522
8.278	19.432
8.333	19.343
8.389	19.254
8.444	19.165
8.5	19.075
8.556	18.986
8.611	18.897
8.667	18.808
8.722	18.718
8.778	18.629
8.833	18.54
8.889	18.451
8.944	18.361
9	18.272
9.056	18.183
9.111	18.094
9.167	18.004
9.222	17.915
9.278	17.826
9.333	17.737
9.389	17.648
9.444	17.558
9.5	17.469
9.556	17.38
9.611	17.291
9.667	17.201
9.722	17.112
9.778	17.023
9.833	16.934
9.889	16.844
9.944	16.755
10	16.666
10.056	16.577
10.111	16.487
10.167	16.398
10.222	16.309
10.278	16.22
10.333	16.13
10.389	16.041
10.444	15.952
10.5	15.863

10.556	15.774
10.611	15.684
10.667	15.595
10.722	15.506
10.778	15.417
10.833	15.327
10.889	15.238
10.944	15.149
11	15.06
11.056	14.97
11.111	14.881
11.167	14.792
11.222	14.703
11.278	14.613
11.333	14.524
11.389	14.435
11.444	14.346
11.5	14.256
11.556	14.167
11.611	14.078
11.667	13.989
11.722	13.899
11.778	13.81
11.833	13.721
11.889	13.632
11.944	13.543
12	13.453
12.056	13.364
12.111	13.275
12.167	13.186
12.222	13.096
12.278	13.007
12.333	12.918
12.389	12.829
12.444	12.739
12.5	12.65
12.556	12.561
12.611	12.472
12.667	12.382
12.722	12.293
12.778	12.204
12.833	12.115
12.889	12.025
12.944	11.936
13	11.847
13.056	11.758
13.111	11.668
13.167	11.579
13.222	11.49
13.278	11.401
13.333	11.312
13.389	11.222
13.444	11.133
13.5	11.044
13.556	10.955
13.611	10.865
13.667	10.776

13.722	10.687
13.778	10.598
13.833	10.508
13.889	10.419
13.944	10.33
14	10.241
14.056	10.151
14.111	10.062
14.167	9.973
14.222	9.884
14.278	9.794
14.333	9.705
14.389	9.616
14.444	9.527
14.5	9.437
14.556	9.348
14.611	9.259
14.667	9.17
14.722	9.081
14.778	8.991
14.833	8.902
14.889	8.813
14.944	8.724
15	8.634
15.056	8.522
15.111	8.363
15.167	8.158
15.222	7.907
15.278	7.609
15.333	7.266
15.389	6.875
15.444	6.439
15.5	5.956
15.556	5.427
15.611	4.852
15.667	4.23
15.722	3.562
15.778	2.848
15.833	2.087
15.889	1.281
15.944	0.427
16	−0.472
16.056	−1.418
16.111	−2.41
16.167	−3.448
16.222	−4.533
16.278	−5.664
16.333	−6.841
16.389	−8.064
16.444	−9.334
16.5	−10.65
16.556	−12.013
16.611	−13.421
16.667	−14.876
16.722	−16.377
16.778	−17.925

16.833	−19.519
16.889	−21.159
16.944	−22.846
17	−24.578
17.056	−26.357
17.111	−28.183
17.167	−30.054
17.222	−31.972
17.278	−33.937
17.333	−35.947
17.389	−38.004
17.444	−40.107
17.5	−42.256
17.556	−44.452
17.611	−46.694
17.667	−48.982
17.722	−51.317
17.778	−53.698
17.833	−56.125
17.889	−58.599
17.944	−61.119
18	−63.685
18.056	−66.297
18.111	−68.956
18.167	−71.661
18.222	−74.412
18.278	−77.21
18.333	−80.053
18.389	−82.944
18.444	−85.88
18.5	−88.863
18.556	−91.892
18.611	−94.967
18.667	−98.089
18.722	−101.257
18.778	−104.471
18.833	−107.732
18.889	−111.038
18.944	−114.391
19	−117.791
19.056	−121.237
19.111	−124.729
19.167	−128.267
19.222	−131.852
19.278	−135.482
19.333	−139.16
19.389	−142.883
19.444	−146.653
19.5	−150.469
19.556	−154.331
19.611	−158.24
19.667	−162.195
19.722	−166.196
19.778	−170.244
19.833	−174.338
19.889	−178.478
19.944	−182.664

20	−186.897
20.056	−183.137
20.111	−179.424
20.167	−175.756
20.222	−172.135
20.278	−168.561
20.333	−165.032
20.389	−161.55
20.444	−158.114
20.5	−154.725
20.556	−151.382
20.611	−148.085
20.667	−144.834
20.722	−141.63
20.778	−138.472
20.833	−135.36
20.889	−132.294
20.944	−129.275
21	−126.302
21.056	−123.376
21.111	−120.496
21.167	−117.662
21.222	−114.874
21.278	−112.132
21.333	−109.437
21.389	−106.789
21.444	−104.186
21.5	−101.63
21.556	−99.12
21.611	−96.656
21.667	−94.239
21.722	−91.868
21.778	−89.543
21.833	−87.265
21.889	−85.033
21.944	−82.847
22	−80.708
22.056	−78.591
22.111	−76.475
22.167	−74.358
22.222	−72.242
22.278	−70.126
22.333	−68.009
22.389	−65.893
22.444	−63.776
22.5	−61.66
22.556	−59.544
22.611	−57.427
22.667	−55.311
22.722	−53.195
22.778	−51.078
22.833	−48.962
22.889	−46.845
22.944	−44.729
23	−42.613
23.056	−40.496

23.111	−38.38
23.167	−36.263
23.222	−34.147
23.278	−32.031
23.333	−29.914
23.389	−27.798
23.444	−25.682
23.5	−23.565
23.556	−21.449
23.611	−19.332
23.667	−17.216
23.722	−15.1
23.778	−12.983
23.833	−10.867
23.889	−8.751
23.944	−6.634
24	−4.518
24.056	−2.401
24.111	−0.285
24.167	1.831
24.222	3.948
24.278	6.064
24.333	8.181
24.389	10.297
24.444	12.413
24.5	14.53
24.556	16.646
24.611	18.762
24.667	20.879
24.722	22.995
24.778	25.112
24.833	27.228
24.889	29.344
24.944	31.461
25	33.577
25.056	35.678
25.111	37.748
25.167	39.787
25.222	41.796
25.278	43.773
25.333	45.72
25.389	47.636
25.444	49.52
25.5	51.375
25.556	53.198
25.611	54.99
25.667	56.751
25.722	58.482
25.778	60.182
25.833	61.851
25.889	63.489
25.944	65.096
26	66.672
26.056	68.217
26.111	69.732
26.167	71.216
26.222	72.668

26.222	[illegible]
26.278	74.09
26.333	75.481
26.389	76.842
26.444	78.171
26.5	79.469
26.556	80.737
26.611	81.974
26.667	83.18
26.722	84.355
26.778	85.499
26.833	86.612
26.889	87.695
26.944	88.746
27	89.767
27.056	90.757
27.111	91.716
27.167	92.644
27.222	93.541
27.278	94.407
27.333	95.243
27.389	96.048
27.444	96.821
27.5	97.564
27.556	98.276
27.611	98.957
27.667	99.608
27.722	100.227
27.778	100.816
27.833	101.374
27.889	101.901
27.944	102.397
28	102.862
28.056	103.311
28.111	103.761
28.167	104.211
28.222	104.661
28.278	105.11
28.333	105.56
28.389	106.01
28.444	106.459
28.5	106.909
28.556	107.359
28.611	107.809
28.667	108.258
28.722	108.708
28.778	109.158
28.833	109.607
28.889	110.057
28.944	110.507
29	110.957
29.056	111.406
29.111	111.856
29.167	112.306
29.222	112.755
29.278	113.205
29.333	113.655

29.389	114.105
29.444	114.554
29.5	115.004
29.556	115.454
29.611	115.903
29.667	116.353
29.722	116.803
29.778	117.253
29.833	117.702
29.889	118.152
29.944	118.602
30	119.051
30.056	118.39
30.111	117.729
30.167	117.067
30.222	116.406
30.278	115.744
30.333	115.083
30.389	114.422
30.444	113.76
30.5	113.099
30.556	112.437
30.611	111.776
30.667	111.115
30.722	110.453
30.778	109.792
30.833	109.13
30.889	108.469
30.944	107.808
31	107.146
31.056	106.485
31.111	105.823
31.167	105.162
31.222	104.501
31.278	103.839
31.333	103.178
31.389	102.516
31.444	101.855
31.5	101.194
31.556	100.532
31.611	99.871
31.667	99.209
31.722	98.548
31.778	97.887
31.833	97.225
31.889	96.564
31.944	95.903
32	95.241
32.056	94.58
32.111	93.918
32.167	93.257
32.222	92.596
32.278	91.934
32.333	91.273
32.389	90.611
32.444	89.95
32.5	89.289

32.5	89.289
32.556	88.627
32.611	87.966
32.667	87.304
32.722	86.643
32.778	85.982
32.833	85.32
32.889	84.659
32.944	83.997
33	83.336
33.056	82.675
33.111	82.013
33.167	81.352
33.222	80.69
33.278	80.029
33.333	79.368
33.389	78.706
33.444	78.045
33.5	77.383
33.556	76.722
33.611	76.061
33.667	75.399
33.722	74.738
33.778	74.076
33.833	73.415
33.889	72.754
33.944	72.092
34	71.431
34.056	70.769
34.111	70.108
34.167	69.447
34.222	68.785
34.278	68.124
34.333	67.462
34.389	66.801
34.444	66.14
34.5	65.478
34.556	64.817
34.611	64.155
34.667	63.494
34.722	62.833
34.778	62.171
34.833	61.51
34.889	60.848
34.944	60.187
35	59.526
35.056	58.864
35.111	58.203
35.167	57.542
35.222	56.88
35.278	56.219
35.333	55.557
35.389	54.896
35.444	54.235
35.5	53.573
35.556	52.912
35.611	52.25

35.667	51.589
35.722	50.928
35.778	50.266
35.833	49.605
35.889	48.943
35.944	48.282
36	47.621
36.056	46.959
36.111	46.298
36.167	45.636
36.222	44.975
36.278	44.314
36.333	43.652
36.389	42.991
36.444	42.329
36.5	41.668
36.556	41.007
36.611	40.345
36.667	39.684
36.722	39.022
36.778	38.361
36.833	37.7
36.889	37.038
36.944	36.377
37	35.715
37.056	35.054
37.111	34.393
37.167	33.731
37.222	33.07
37.278	32.408
37.333	31.747
37.389	31.086
37.444	30.424
37.5	29.763
37.556	29.101
37.611	28.44
37.667	27.779
37.722	27.117
37.778	26.456
37.833	25.794
37.889	25.133
37.944	24.472
38	23.81
38.056	23.149
38.111	22.487
38.167	21.826
38.222	21.165
38.278	20.503
38.333	19.842
38.389	19.181
38.444	18.519
38.5	17.858
38.556	17.196
38.611	16.535
38.667	15.874
38.722	15.212

38.778	14.551
38.833	13.889
38.889	13.228
38.944	12.567
39	11.905
39.056	11.244
39.111	10.582
39.167	9.921
39.222	9.26
39.278	8.598
39.333	7.937
39.389	7.275
39.444	6.614
39.5	5.953
39.556	5.291
39.611	4.63
39.667	3.968
39.722	3.307
39.778	2.646
39.833	1.984
39.889	1.323
39.944	0.661
40	0

0	0
0	0
0.056	−0.0263
0.111	−0.1045
0.167	−0.2333
0.222	−0.4115
0.278	−0.6378
0.333	−0.9109
0.389	−1.2297
0.444	−1.5929
0.5	−1.9992
0.556	−2.4474
0.611	−2.9362
0.667	−3.4644
0.722	−4.0308
0.778	−4.6341
0.833	−5.2731
0.889	−5.9465
0.944	−6.6531
1	−7.3916
1.056	−8.1608
1.111	−8.9594
1.167	−9.7862
1.222	−10.64
1.278	−11.5195
1.333	−12.4234
1.389	−13.3506
1.444	−14.2997
1.5	−15.2695
1.556	−16.2589
1.611	−17.2664
1.667	−18.291
1.722	−19.3313
1.778	−20.3861

1.833	−21.4542
1.889	−22.5342
1.944	−23.6251
2	−24.7254
2.056	−25.8342
2.111	−26.9509
2.167	−28.0751
2.222	−29.2065
2.278	−30.3447
2.333	−31.4893
2.389	−32.64
2.444	−33.7964
2.5	−34.9581
2.556	−36.1248
2.611	−37.2961
2.667	−38.4716
2.722	−39.6509
2.778	−40.8338
2.833	−42.0197
2.889	−43.2085
2.944	−44.3996
3	−45.5927
3.056	−46.7875
3.111	−47.9836
3.167	−49.1805
3.222	−50.378
3.278	−51.5757
3.333	−52.7733
3.389	−53.9702
3.444	−55.1662
3.5	−56.3609
3.556	−57.554
3.611	−58.745
3.667	−59.9337
3.722	−61.1195
3.778	−62.3023
3.833	−63.4815
3.889	−64.6569
3.944	−65.828
4	−66.9945
4.056	−68.156
4.111	−69.3122
4.167	−70.4627
4.222	−71.6071
4.278	−72.745
4.333	−73.8762
4.389	−75.0001
4.444	−76.1165
4.5	−77.225
4.556	−78.3252
4.611	−79.4167
4.667	−80.4992
4.722	−81.5723
4.778	−82.6357
4.833	−83.6889
4.889	−84.7317

4.944	−85.7635
5	−86.7842
5.056	−87.7932
5.111	−88.7903
5.167	−89.775
5.222	−90.747
5.278	−91.706
5.333	−92.6515
5.389	−93.5832
5.444	−94.5007
5.5	−95.4037
5.556	−96.2917
5.611	−97.1645
5.667	−98.0216
5.722	−98.8627
5.778	−99.6875
5.833	−100.4955
5.889	−101.2863
5.944	−102.0597
6	−102.8152
6.056	−103.5525
6.111	−104.2717
6.167	−104.9728
6.222	−105.6559
6.278	−106.3211
6.333	−106.9684
6.389	−107.5978
6.444	−108.2096
6.5	−108.8037
6.556	−109.3802
6.611	−109.9391
6.667	−110.4806
6.722	−111.0048
6.778	−111.5116
6.833	−112.0012
6.889	−112.4736
6.944	−112.9289
7	−113.3672
7.056	−113.7885
7.111	−114.193
7.167	−114.5806
7.222	−114.9515
7.278	−115.3057
7.333	−115.6433
7.389	−115.9643
7.444	−116.2689
7.5	−116.5571
7.556	−116.829
7.611	−117.0846
7.667	−117.3241
7.722	−117.5474
7.778	−117.7547
7.833	−117.946
7.889	−118.1214
7.944	−118.281
8	−118.4248
8.056	−118.553

8.111	−118.6655
8.167	−118.7624
8.222	−118.8439
8.278	−118.91
8.333	−118.9607
8.389	−118.9962
8.444	−119.0164
8.5	−119.0216
8.556	−119.0117
8.611	−118.9868
8.667	−118.947
8.722	−118.8923
8.778	−118.8229
8.833	−118.7387
8.889	−118.64
8.944	−118.5266
9	−118.3988
9.056	−118.2566
9.111	−118.1
9.167	−117.9291
9.222	−117.744
9.278	−117.5447
9.333	−117.3314
9.389	−117.1041
9.444	−116.8629
9.5	−116.6078
9.556	−116.3389
9.611	−116.0563
9.667	−115.7601
9.722	−115.4503
9.778	−115.1269
9.833	−114.7902
9.889	−114.44
9.944	−114.0766
10	−113.6999
10.056	−113.3101
10.111	−112.9072
10.167	−112.4913
10.222	−112.0625
10.278	−111.6208
10.333	−111.1662
10.389	−110.699
10.444	−110.2191
10.5	−109.7266
10.556	−109.2215
10.611	−108.7041
10.667	−108.1742
10.722	−107.632
10.778	−107.0776
10.833	−106.5111
10.889	−105.9324
10.944	−105.3417
11	−104.739
11.056	−104.1245
11.111	−103.4981
11.167	−102.86

11.222	−102.2102
11.278	−101.5489
11.333	−100.8759
11.389	−100.1916
11.444	−99.4958
11.5	−98.7887
11.556	−98.0703
11.611	−97.3408
11.667	−96.6001
11.722	−95.8485
11.778	−95.0858
11.833	−94.3122
11.889	−93.5278
11.944	−92.7327
12	−91.9269
12.056	−91.1104
12.111	−90.2834
12.167	−89.4459
12.222	−88.598
12.278	−87.7398
12.333	−86.8713
12.389	−85.9926
12.444	−85.1038
12.5	−84.2049
12.556	−83.2961
12.611	−82.3773
12.667	−81.4487
12.722	−80.5103
12.778	−79.5622
12.833	−78.6045
12.889	−77.6372
12.944	−76.6604
13	−75.6742
13.056	−74.6787
13.111	−73.6738
13.167	−72.6598
13.222	−71.6366
13.278	−70.6044
13.333	−69.5631
13.389	−68.5129
13.444	−67.4539
13.5	−66.3861
13.556	−65.3096
13.611	−64.2244
13.667	−63.1306
13.722	−62.0283
13.778	−60.9176
13.833	−59.7986
13.889	−58.6712
13.944	−57.5356
14	−56.3919
14.056	−55.2401
14.111	−54.0802
14.167	−52.9125
14.222	−51.7368
14.278	−50.5534
14.333	−49.3622

14.333	−49.3622
14.389	−48.1634
14.444	−46.9569
14.5	−45.743
14.556	−44.5216
14.611	−43.2928
14.667	−42.0567
14.722	−40.8134
14.778	−39.5629
14.833	−38.3053
14.889	−37.0407
14.944	−35.7691
15	−34.4907
15.056	−33.2054
15.111	−31.9134
15.167	−30.6149
15.222	−29.3099
15.278	−27.9986
15.333	−26.6814
15.389	−25.3584
15.444	−24.03
15.5	−22.6966
15.556	−21.3585
15.611	−20.016
15.667	−18.6698
15.722	−17.3202
15.778	−15.9679
15.833	−14.6133
15.889	−13.257
15.944	−11.8998
16	−10.5422
16.056	−9.185
16.111	−7.8289
16.167	−6.4748
16.222	−5.1233
16.278	−3.7755
16.333	−2.4322
16.389	−1.0942
16.444	0.2374
16.5	1.5616
16.556	2.8774
16.611	4.1837
16.667	5.4793
16.722	6.7633
16.778	8.0343
16.833	9.2911
16.889	10.5326
16.944	11.7573
17	12.964
17.056	14.1512
17.111	15.3177
17.167	16.4619
17.222	17.5824
17.278	18.6776
17.333	19.746
17.389	20.7861
17.444	21.7961

17.5	22.7745
17.556	23.7195
17.611	24.6294
17.667	25.5025
17.722	26.3369
17.778	27.1307
17.833	27.8822
17.889	28.5894
17.944	29.2503
18	29.8629
18.056	30.4253
18.111	30.9354
18.167	31.391
18.222	31.79
18.278	32.1303
18.333	32.4096
18.389	32.6258
18.444	32.7765
18.5	32.8593
18.556	32.8721
18.611	32.8122
18.667	32.6775
18.722	32.4653
18.778	32.1731
18.833	31.7985
18.889	31.3388
18.944	30.7915
19	30.1539
19.056	29.4233
19.111	28.597
19.167	27.6723
19.222	26.6463
19.278	25.5162
19.333	24.2792
19.389	22.9323
19.444	21.4727
19.5	19.8972
19.556	18.203
19.611	16.387
19.667	14.4461
19.722	12.3771
19.778	10.177
19.833	7.8424
19.889	5.3703
19.944	2.7572
20	0
20.056	−2.9206
20.111	−6.0192
20.167	−9.2922
20.222	−12.736
20.278	−16.3471
20.333	−20.1221
20.389	−24.0574
20.444	−28.1497
20.5	−32.3957
20.556	−36.7921

20.611	−41.3356
20.667	−46.023
20.722	−50.8512
20.778	−55.817
20.833	−60.9175
20.889	−66.1494
20.944	−71.51
21	−76.9962
21.056	−82.6051
21.111	−88.334
21.167	−94.18
21.222	−100.1403
21.278	−106.2123
21.333	−112.3932
21.389	−118.6805
21.444	−125.0716
21.5	−131.564
21.556	−138.1552
21.611	−144.8427
21.667	−151.6241
21.722	−158.4971
21.778	−165.4594
21.833	−172.5088
21.889	−179.6429
21.944	−186.8597
22	−194.1571
22.056	−201.5329
22.111	−208.985
22.167	−216.5115
22.222	−224.1102
22.278	−231.7791
22.333	−239.5162
22.389	−247.3194
22.444	−255.1866
22.5	−263.1158
22.556	−271.1049
22.611	−279.1519
22.667	−287.2547
22.722	−295.4112
22.778	−303.6195
22.833	−311.8774
22.889	−320.1828
22.944	−328.5338
23	−336.9283
23.056	−345.3641
23.111	−353.8393
23.167	−362.3519
23.222	−370.8996
23.278	−379.4806
23.333	−388.0926
23.389	−396.7338
23.444	−405.4019
23.5	−414.095
23.556	−422.8111
23.611	−431.5479
23.667	−440.3036
23.722	−449.076

23.778	−457.863
23.833	−466.6627
23.889	−475.4729
23.944	−484.2916
24	−493.1168
24.056	−501.9464
24.111	−510.7783
24.167	−519.6105
24.222	−528.4409
24.278	−537.2675
24.333	−546.0881
24.389	−554.9009
24.444	−563.7036
24.5	−572.4942
24.556	−581.2708
24.611	−590.0311
24.667	−598.7732
24.722	−607.4951
24.778	−616.1946
24.833	−624.8696
24.889	−633.5183
24.944	−642.1384
25	−650.7279
25.056	−659.2848
25.111	−667.807
25.167	−676.2926
25.222	−684.7395
25.278	−693.1458
25.333	−701.5096
25.389	−709.8289
25.444	−718.102
25.5	−726.3269
25.556	−734.5019
25.611	−742.6252
25.667	−750.6951
25.722	−758.7099
25.778	−766.6678
25.833	−774.5672
25.889	−782.4066
25.944	−790.1842
26	−797.8986
26.056	−805.5482
26.111	−813.1315
26.167	−820.6471
26.222	−828.0935
26.278	−835.4692
26.333	−842.773
26.389	−850.0034
26.444	−857.1591
26.5	−864.2389
26.556	−871.2415
26.611	−878.1656
26.667	−885.01
26.722	−891.7736
26.778	−898.4553
26.833	−905.0539

26.889	−911.5683
26.944	−917.9975
27	−924.3404
27.056	−930.5961
27.111	−936.7637
27.167	−942.8421
27.222	−948.8305
27.278	−954.728
27.333	−960.5337
27.389	−966.2469
27.444	−971.8668
27.5	−977.3925
27.556	−982.8235
27.611	−988.159
27.667	−993.3983
27.722	−998.5408
27.778	−1003.5859
27.833	−1008.5331
27.889	−1013.3817
27.944	−1018.1313
28	−1022.7815
28.056	−1027.3316
28.111	−1031.7814
28.167	−1036.1303
28.222	−1040.378
28.278	−1044.5239
28.333	−1048.5677
28.389	−1052.5089
28.444	−1056.3471
28.5	−1060.0818
28.556	−1063.7127
28.611	−1067.2391
28.667	−1070.6609
28.722	−1073.9774
28.778	−1077.1883
28.833	−1080.2931
28.889	−1083.2913
28.944	−1086.1827
29	−1088.9666
29.056	−1091.6427
29.111	−1094.2105
29.167	−1096.6697
29.222	−1099.0196
29.278	−1101.2601
29.333	−1103.3904
29.389	−1105.4104
29.444	−1107.3194
29.5	−1109.1172
29.556	−1110.8031
29.611	−1112.3769
29.667	−1113.838
29.722	−1115.1861
29.778	−1116.4206
29.833	−1117.5412
29.889	−1118.5474
29.944	−1119.4388
30	−1120.215

30.056	−1120.8756
30.111	−1121.4211
30.167	−1121.8523
30.222	−1122.1697
30.278	−1122.3739
30.333	−1122.4657
30.389	−1122.4456
30.444	−1122.3144
30.5	−1122.0726
30.556	−1121.7208
30.611	−1121.2598
30.667	−1120.6902
30.722	−1120.0126
30.778	−1119.2277
30.833	−1118.336
30.889	−1117.3383
30.944	−1116.2352
31	−1115.0273
31.056	−1113.7153
31.111	−1112.2998
31.167	−1110.7815
31.222	−1109.161
31.278	−1107.4389
31.333	−1105.6159
31.389	−1103.6926
31.444	−1101.6697
31.5	−1099.5479
31.556	−1097.3276
31.611	−1095.0097
31.667	−1092.5947
31.722	−1090.0834
31.778	−1087.4762
31.833	−1084.7739
31.889	−1081.9771
31.944	−1079.0865
32	−1076.1027
32.056	−1073.0263
32.111	−1069.858
32.167	−1066.5985
32.222	−1063.2483
32.278	−1059.8081
32.333	−1056.2786
32.389	−1052.6604
32.444	−1048.9541
32.5	−1045.1604
32.556	−1041.28
32.611	−1037.3134
32.667	−1033.2613
32.722	−1029.1244
32.778	−1024.9033
32.833	−1020.5986
32.889	−1016.211
32.944	−1011.7411
33	−1007.1896
33.056	−1002.5571
33.111	−997.8443

33.167	−993.0518
33.222	−988.1802
33.278	−983.2302
33.333	−978.2024
33.389	−973.0975
33.444	−967.9161
33.5	−962.6588
33.556	−957.3264
33.611	−951.9194
33.667	−946.4384
33.722	−940.8842
33.778	−935.2574
33.833	−929.5586
33.889	−923.7884
33.944	−917.9475
34	−912.0366
34.056	−906.0563
34.111	−900.0071
34.167	−893.8899
34.222	−887.7051
34.278	−881.4536
34.333	−875.1358
34.389	−868.7524
34.444	−862.3041
34.5	−855.7916
34.556	−849.2154
34.611	−842.5762
34.667	−835.8747
34.722	−829.1115
34.778	−822.2872
34.833	−815.4025
34.889	−808.458
34.944	−801.4544
35	−794.3923
35.056	−787.2723
35.111	−780.0952
35.167	−772.8614
35.222	−765.5718
35.278	−758.2269
35.333	−750.8273
35.389	−743.3738
35.444	−735.8669
35.5	−728.3073
35.556	−720.6956
35.611	−713.0325
35.667	−705.3187
35.722	−697.5547
35.778	−689.7412
35.833	−681.8789
35.889	−673.9683
35.944	−666.0102
36	−658.0052
36.056	−649.9539
36.111	−641.8569
36.167	−633.715
36.222	−625.5287
36.278	−617.2987

36.333	−609.0256
36.389	−600.7101
36.444	−592.3529
36.5	−583.9545
36.556	−575.5156
36.611	−567.0369
36.667	−558.5189
36.722	−549.9624
36.778	−541.368
36.833	−532.7362
36.889	−524.0679
36.944	−515.3635
37	−506.6238
37.056	−497.8494
37.111	−489.0409
37.167	−480.199
37.222	−471.3244
37.278	−462.4175
37.333	−453.4792
37.389	−444.5101
37.444	−435.5107
37.5	−426.4817
37.556	−417.4239
37.611	−408.3377
37.667	−399.2239
37.722	−390.0832
37.778	−380.916
37.833	−371.7232
37.889	−362.5053
37.944	−353.2629
38	−343.9968
38.056	−334.7076
38.111	−325.3958
38.167	−316.0622
38.222	−306.7074
38.278	−297.332
38.333	−287.9367
38.389	−278.5221
38.444	−269.0888
38.5	−259.6376
38.556	−250.169
38.611	−240.6837
38.667	−231.1823
38.722	−221.6656
38.778	−212.134
38.833	−202.5883
38.889	−193.029
38.944	−183.457
39	−173.8727
39.056	−164.2768
39.111	−154.67
39.167	−145.053
39.222	−135.4263
39.278	−125.7906
39.333	−116.1465
39.389	−106.4947

39.444	−96.8359
39.5	−87.1706
39.556	−77.4996
39.611	−67.8234
39.667	−58.1427
39.722	−48.4581
39.778	−38.7704
39.833	−29.08
39.889	−19.3877
39.944	−9.6942
40	0

0
2
6
15
20
22
25
28
30
40