

Калькулятор SplineCalc.

Версия 6.0. 01.10.2015г.

Руководство пользователя.

Содержание.

1. Введение. Главное окно диалога.
2. Типы аргументов функций.
3. Загрузка файлов.
4. Ввод и сохранение строк в таблю.
5. Функции Калькулятора.
6. Визуализация и сохранение результатов.

1. Введение. Главное окно диалога.

Калькулятор **SplineCalc** выполняет математические операции с картами (гридами), таблицами данных, линиями, полигонами и числами.



Главное окно Калькулятора включает в себя :

блок загрузки файлов - аргументов;

табло - многострочное редактируемое окно - для отображения вводимой информации;

поле сообщений - для информации о типах вводимых функций, вывода результата и сообщений об ошибке;

кнопки управления:

«Сброс» - для очистки табло и поля сообщений,

«<-» - для удаления из табло последнего символа,

«Расчет» - для начала вычислений,

«Сохранить» - для сохранения результата в файле,

«Вид» - для графического представления результата в отдельном окне,

«>>», «<<» - для сохранения в файле и загрузки из файла содержимого табло,

«ID» - для отображения значений идентификаторов,

«м – км» - для переключения масштаба линейных аргументов;

кнопки ввода в табло констант и математических функций;

кнопки ввода чисел, знаков операций и пунктуации;

кнопки помощи и истории ввода в табло.

Калькулятор позволяет :

загружать файлы гридов, линий, полигонов и таблиц данных для последующих операций с ними;

вводить в табло наборы символов и проводить их синтаксический анализ;

вычислять широкий набор математических функций с аргументами пяти типов;

обрабатывать ошибки ввода и вычислений;

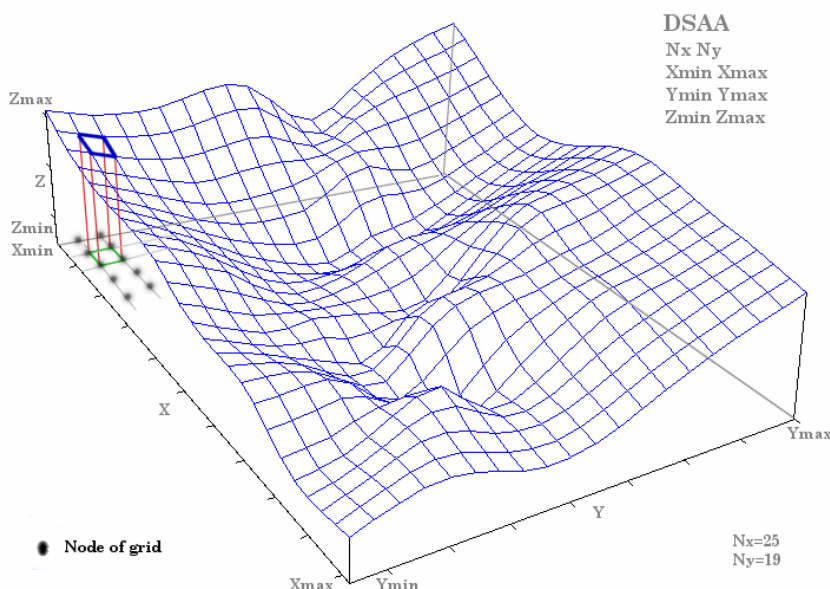
сохранять в файлах содержимое табло, конечный и промежуточные результаты;

отображать результаты расчета в графическом окне.

2. Типы аргументов функций.

Аргументы функций Калькулятора **SplineCalc** имеют типы КАРТА, ДАННЫЕ, ЛИНИИ, ПОЛИГОН, ЧИСЛО. Они имеют определенное математическое представление, которое вытекает из сути решаемых задач (например, геологических).

Аргумент типа КАРТА соответствует файлу формата программы Surfer, имеющему расширение **grd**, который содержит значения поверхности (карты) в узлах прямоугольной сетки (грида).



Ниже представлен пример записи в файле грида.

DSAA

11 5

1500 2500

77000 77800

24.0 35.6

24.1 24.2 24.8 26. 27.0 29. 29.99 31.0 31 33 32

25.0 25.5 25.7 28. 28.1 29. 29.7 30.8 33. 33 31

27.7 28.0 29.0 24.0 31.5 32. 34.0 34.4 35.6 34 30

26.0 25.0 24.7 24.4 24.3 24.8 24.9 25.0 26. 30 28

28.2 27.0 26.0 25. 24.9 24.4 24.8 27.0 28.3 31 30

Первая строка - ключевое слово грида.

Вторая строка - два целых числа - количество узлов сетки по осям X и Y, соответственно.

Третья строка - два числа, задающие размеры прямоугольной области грида по оси X (минимальное и максимальное значения).

Четвертая строка - аналогично предыдущей, по оси Y.

Пятая строка - два числа, определяющие минимальное и максимальное значения карты в узлах сетки.

Шестая и последующие строки - значения карты в узлах грида; порядок записи - с левого нижнего узла направо.

Для определения аргумента типа ДАННЫЕ требуется текстовый файл (обычно с расширением **dat**), содержащий набор строк и столбцов данных в точках (каждая строка файла соответствует одной точке). Информация о точках (в частности, их координаты) может содержаться в любом столбце. Не допускаются разрывы информации в каком-либо столбце. В столбцах координат не допускаются буквенные символы. Допускается вырождение таблицы до числа, то есть таблица может содержать один столбец и одну строку. Пример таблицы:

101 13313 12077 1 2441 16.8 16.8 16.8

103 13816 11814 1 2444 28.8 17.4 28.8

```

106 14015 12951 1 2439 8.2 8.2 8.2
107 14262 12278 1 2434 26.0 26.0 26.0
107с 14436 11789 1 2462 9.4 3.0 9.4

```

Аргументом типа ДАННЫЕ, который участвует в расчетах, является столбец файла. Однако большинство функций требует совокупность трех столбцов: собственно, данные в точках (столбец Z) и координаты точек (столбцы X и Y). Столбец Z может совпадать с одним из столбцов X и Y, поэтому файл данных может содержать произвольное количество столбцов.

ДАННЫЕ могут храниться также в shape-файлах.

Аргументами типов ЛИНИИ и ПОЛИГОН служит полилиния - одна или несколько ломаных линий (многозвенников), при этом ПОЛИГОН содержит только взаимно непересекающиеся замкнутые линии. Для хранения в файлах используются известные форматы: Shape ArcView (в бинарном виде, расширение **shp**) и ArcInfo (ASCII, расширение **con** или **lin, pol**).

В формате ArcInfo полилинии имеют следующее представление (пример):

```

0
13583.333 12250
13916.667 12250
END
1
13916.667 12666.667
13583.333 12250
13583.333 12666.667
END
2
13916.667 12250
14250 12666.667
14250 12250
13916.667 12250
END
END

```

Каждый многозвенник имеет в записи идентификатор - целое число, координаты (X, Y) узлов и завершающее слово END. Такое же слово добавляется в конце файла.

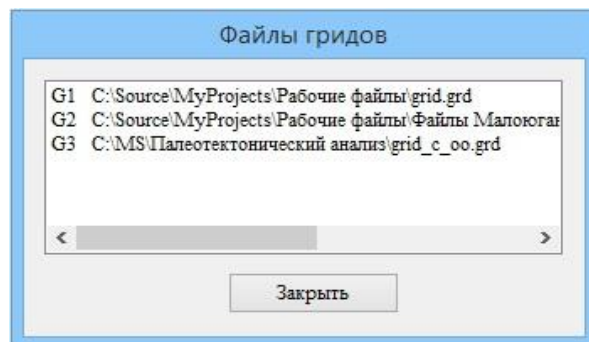
ПОЛИГОН, в зависимости от количества, взаимного расположения и уровня вложенности друг в друга замкнутых линий, может быть достаточно сложным объектом, имеющим «дырки», каждая из которых может содержать «островки» и т.д. Это свойство полигонов используется в функциях их попарного объединения (сложения), пересечения (умножения) и вычитания.

ЧИСЛО – любое целое или дробное, записанное как в виде с плавающей точкой, так и в экспоненциальной форме.

3. Загрузка файлов.

Для загрузки аргументов, являющихся файлами карт, линий, полигонов и данных, служат соответствующие кнопки в главном окне Калькулятора.

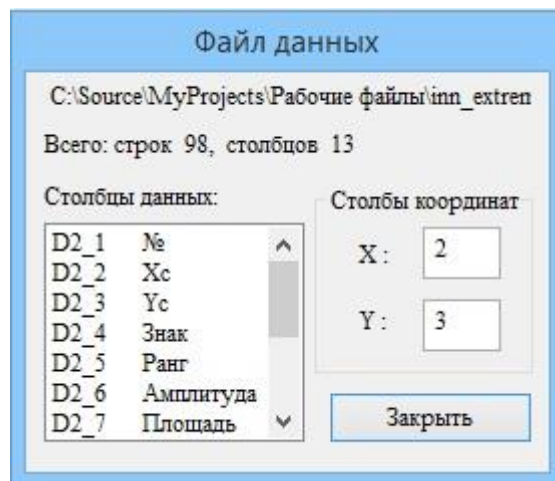
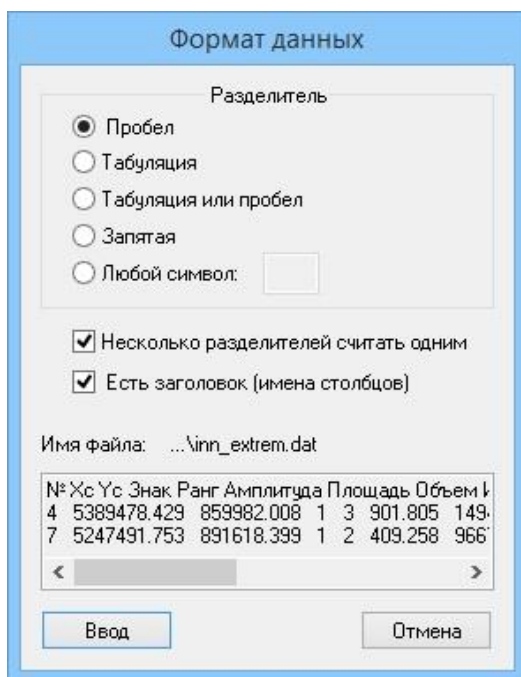
При загрузке очередной карты кнопкой «Гриды» открывается стандартное окно выбора файла. Путь выбранного файла заносится в список окна «Файлы гридов». Новому файлу присваивается очередной идентификатор, начинающийся с буквы **Г**. Последний (при селектировании строки списка) будет вводиться в табло и служить аргументом типа КАРТА. Пополнение списка карт возможно на любом этапе вычислений. При закрытии окна списка карт значения идентификаторов сохраняются в памяти Калькулятора до обновления списка (загрузки новых гридов).



Аналогично загружаются файлы линий и полигонов, отличие состоит лишь в названии идентификаторов: для линий они начинаются с буквы **С**, для полигонов - с буквы **Р**. При загрузке share-файлов появится дополнительное окно «DBF - файл», в котором следует указать, из какого столбца загружать значения аргумента.

Кнопкой «Данные» аналогично загружаются файлы данных, но, в отличие от предыдущих типов, для каждого введенного файла появляется отдельное окно «Файл данных». В этом окне содержатся: полный путь файла, количество строк и столбцов, поля ввода номеров столбцов координат точек **X** и **Y** (по умолчанию, 1 и 2), а также список пронумерованных столбцов файла. Каждому столбцу соответствует строка списка, начинающаяся идентификатором **Di_j** (i - индекс окна данных, j - номер столбца), который при посылке его в табло служит аргументом типа ДАННЫЕ. Если хотя бы одно поле столбцов координат пустое, вычисления проводятся без учета координат точек.

При загрузке файла данных сначала появляется окно, в котором следует указать формат файла, а именно: разделитель между столбцами (по умолчанию, пробел), отметить наличие или отсутствие строки заголовка (имена столбцов) в файле.



4. Ввод и сохранение строк в табло.

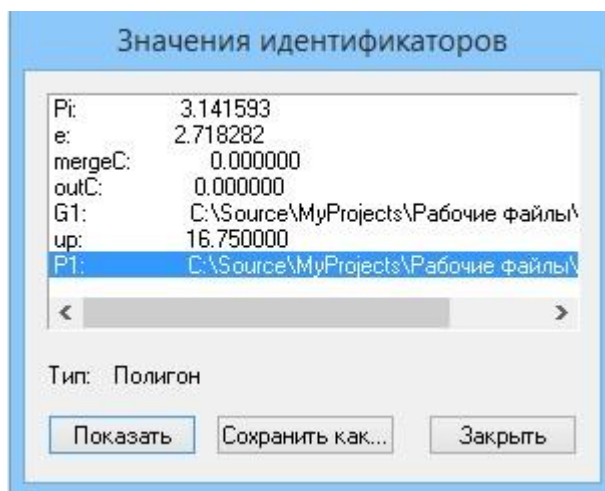
Ввод символов в табло производится как с помощью мыши (кнопками Калькулятора и селектированием строк списков файлов), так и с клавиатуры пользователя. Конкретное место ввода в табло указывается курсором. Каждое математическое выражение должно заканчиваться знаком «;». Программа не накладывает ограничений на количество вводимых строк и вложенность функций. Любое промежуточное выражение может быть отмечено идентификатором, который используется в последующих вычислениях.

Идентификатором должен быть набор символов латинского алфавита, начинающийся с буквы.

Пример правильно введенной в табло строки:

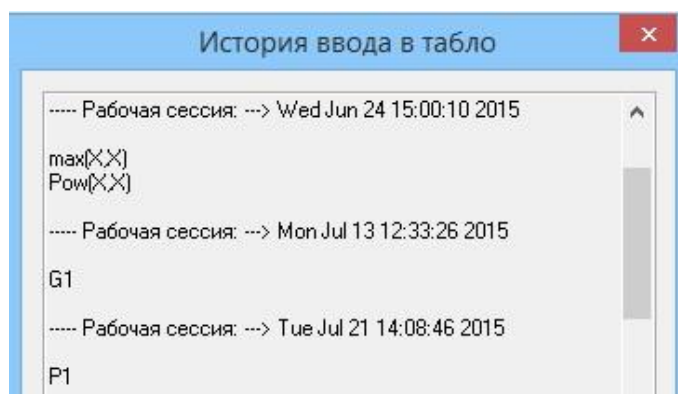
$$p = \min(\text{Abs}(G1), G2 + 220); \text{Integral}(p + \max(G1, 2745.5), P1);$$

Значение идентификаторов сохраняется до конца сеанса работы с программой и может изменяться в ходе вычислений. Текущие значения идентификаторов можно проследить, вызвав их список с помощью кнопки «ID».



Двойной щелчок на строке списка позволяет занести в табло выбранный идентификатор. Нечисловые значения (гриды, линии и т.д.) можно визуализировать и сохранить в файлах.

История ввода строк в табло сохраняется при каждом нажатии кнопки «Сброс». При этом содержимое табло заносится в служебный файл **history.ses** как очередной сеанс работы. Содержимое файла можно посмотреть (и скопировать ранее введенные строки), нажав кнопку «История».



Кроме этого, содержимое табло пользователь может сохранить в файле (с расширением **ctb**), используя кнопку «>>». Соответственно, кнопка «<<» позволяет внести в табло строки, ранее сохраненные в файле.

Применяя такого рода восстановление строк в табло, при дальнейших вычислениях необходимо помнить, что строки, "пришедшие" из других сеансов работы Калькулятора, скорее всего применялись для иных аргументов функций. Поэтому следует проверить, все ли встречающиеся в табло идентификаторы должным образом инициализированы и соответствуют новому расчету.

5. Функции Калькулятора.

Это математические операции с аргументами КАРТА, ДАННЫЕ, ЛИНИИ, ПОЛИГОН, ЧИСЛО. Перечень функций вытекает из необходимости решения практических (в большей степени, геологических) задач.

Условно функции Калькулятора можно разделить на несколько типов: алгебраические (например, **Sqrt**), тригонометрические (например, **ArcSin**), логические (например, **And**) и специальные. Формат записи и допустимые типы аргументов каждой функции можно узнать, вызвав кнопкой «?» окно помощи. Кроме того, при наведении курсора к кнопке соответствующей функции появляется всплывающая подсказка, а при нажатии на кнопку информация о функции отображается в поле сообщений.

Унарные функции (имеющие один аргумент) возвращают тип своего аргумента, а бинарные допускают следующее смешение типов:

- операция с типами ЧИСЛО и КАРТА дает тип КАРТА (каждое значение в узлах грида оперирует с данным числом);
- операция с типами ЧИСЛО и ДАННЫЕ дает тип ДАННЫЕ (каждое значение столбца данных оперирует с числом);
- операция с типами КАРТА и ДАННЫЕ дает тип ДАННЫЕ. Для каждого значения столбца данных проверяется принадлежность карте точки с соответствующими координатами. Если точка - в пределах карты, в операции участвует картируемый параметр, если - вне, в качестве параметра карты выступает значение константы **outC** (по умолчанию, оно равно -9999). Значение этой константы, как и других, может быть переопределено.

Бинарные операции с КАРТАМИ дают в результате КАРТУ, определенную в области пересечения исходных карт. Шаг сетки по каждой оси соответствует шагу более густой сетки карты - аргумента.

Бинарные операции с ДАННЫМИ возвращают ДАННЫЕ, при этом проводится контроль полного совпадения значений координат точек аргументов.

Логические функции возвращают число, столбец чисел или «знаковую» карту, состоящие из 0 и 1.

При выполнении тригонометрических функций используется мера углов, соответствующая положению переключателя «радиан – градус».

Функции **min** и **max** (а также знаки «+», «-», «*», «/»), кроме обычных действий, применяются для изменения числовых значений на линиях (атрибутов). В этом случае один из аргументов бинарных операций должен иметь тип ЛИНИИ или ПОЛИГОН, другой - ЧИСЛО.

Наибольший практический интерес представляют специальные функции. Следующие функции направлены на работу с КАРТОЙ:

EG, XG, YG, RG – функции генерирования гридов, соответственно: единичного, значений, равных координате X или Y, случайных величин. Функции могут принимать параметры, определяющие геометрию нового грида (размеры грида и шаг сетки по осям), или прототип – известный грид. Таким образом, возможна одна из записей:

EG(Xmin, Xmax, Ymin, Ymax, Nx, Ny) или EG(G1).

Функция RG(G, N, a, b, c), кроме того, принимает дополнительные параметры:

- число N, задающее закон распределения непрерывной случайной величины,
- a, b, c - параметры расположения, формы, масштаба случайной величины (b, c вводятся не для всех законов распределения).

Применимы следующие законы распределения с плотностью $f(x)$:

N = 1 – равномерный на отрезке [a, b], $f(x) = 1/(b-a)$,

N = 2 – нормальный, $f(x) = 1/(b*\text{Sqrt}(2*Pi)) * \text{Exp}(-(x-a)^2/(2*b^2))$, $b>0$,

N = 3 – логнормальный, $f(x) = 1/(x*b*\text{Sqrt}(2*Pi)) * \text{Exp}(-(Ln(x)-a)^2/(2*b^2))$, $b>0$,

N = 4 – экспоненциальный, $f(x) = 1/a*\text{Exp}(-x/a)$, $a>0$,

N = 5 – Вейбулла, $f(x) = a/b*(x/b)^{(a-1)}*\text{Exp}(-(x/b)^a)$, $a>0$, $b>0$,

N = 6 – Рэля, $f(x) = x/a^2*\text{Exp}(-x^2/(2*a^2))$, $a>0$,

N = 7 – Парето, $f(x) = a*x^{(-a-1)}$, $a>0$,

N = 8 – треугольный, $f(x) = 2*(x-a)/((b-a)*(c-a))$, $a\leq x < c$,

$f(x) = 2*(b-x)/((b-c)*(b-a))$, $c\leq x \leq b$,

где c - основание проекции вершины треугольника на отрезок [a, b].

Dense(G, N) – сгущение (разрежение) сетки грида G в N раз.

Cut – вырезание фрагмента из грида – по номерам узлов сетки либо по заданному прямоугольнику.

Merge(G1, G2) – слияние (объединение) карт, при котором значение новой карты равно:

- среднему арифметическому - в области пересечения карт,
- значению одной из карт - в области принадлежности только этой карте,
- константе **mergeC** - в области вне карт - аргументов.

Trace-L, Trace-P – функции трассировки, первая из которых формирует изолинии грида, вторая – полигон из линий, расположенных выше или ниже некоторого сечения грида.

Integral(G, P) - интеграл от карты (грида) по замкнутому контуру (полигону); результат - ЧИСЛО.

Dx(G), Dy(G), Dxx(G), Dyy(G), Dxy(G) – дифференцирование грида – создание грида частных и смешанных производных.

Spec(G, N) – специальные дифференциальные функции:

N = 1: $\text{Sqrt} (Dx^2 + Dy^2)$ - уклон (норма градиента);

N = 2: $Dxx^2 + 2. * Dxy^2 + Dyy^2$ - плотность энергии деформации;

N = 3: $\text{ArcTg} (\text{Sqrt} (Dx^2 + Dy^2)) * 180/Pi$ - крутизна склонов (в градусах);

N = 4: $\text{Sqrt} (Dx^2 + Dy^2) / \text{Sqrt}(1 + Dx^2 + Dy^2)$ - фактор крутизны (безразмерный);

N = 5: $\text{sign_Dx}=0$; Если $Dx>0$ $\text{sign_Dx}=1$; если $Dx<0$ $\text{sign_Dx}=-1$;

$\text{sign_Dy}=0$; Если $Dy>0$ $\text{sign_Dy}=1$; если $Dy<0$ $\text{sign_Dy}=-1$;

$-90*(1-\text{sign_Dy})*(1-\text{Abs}(\text{sign_Dx}))+180*(1+\text{sign_Dx})-180*\text{sign_Dx} *$

$\text{ArcCos}(-Dy/\text{Sqrt}(Dx^2 + Dy^2))/PI$ - ориентация склонов (в градусах);

N = 6: $D6 = -(Dxx * (1. + Dy^2) - 2.*Dx*Dy*Dxy + Dyy*(1. + Dx^2)) /$

(2. * Sqrt ((1. + Dx^2 + Dy^2)^3)) - средняя кривизна (1/м);
N = 7: D7 = Sqrt((Dxx*Sqrt((1+Dy^2)/(1+Dx^2))-Dyy/Sqrt((1+Dy^2)/(1+Dx^2))) *
(Dxx*Sqrt((1+Dy^2)/(1+Dx^2))-Dyy/Sqrt((1+Dy^2)/(1+Dx^2)))/(1+Dx^2+Dy^2)+
(Dx*Dy*Dxx*Sqrt((1+Dy^2)/(1+Dx^2))-2*Sqrt((1+Dy^2)*(1+Dx^2))*Dxy+Dx*Dy *
Dyy/Sqrt((1+Dy^2)/(1+Dx^2)))*(Dx*Dy*Dxx*Sqrt((1+Dy^2)/(1+Dx^2))-2*Sqrt((1+
Dy^2)*(1+Dx^2))*Dxy+Dx*Dy*Dyy/Sqrt((1+Dy^2)/(1+Dx^2))))/(2*Sqrt((1+Dx^2+
Dy^2)^3)) - несферичность (1/м);
N = 8: D8 = (Dy^2*Dxx -2*Dx*Dy*Dxy + Dx^2*Dyy)/((Dx^2+ Dy^2)*
Sqrt(1+Dx^2+Dy^2))-((1+Dy^2)*Dxx-2*Dx*Dy*Dxy+(1+Dx^2)* Dyy)/
(2*Sqrt((1+ Dx^2+ Dy^2)^3)) - разностная кривизна (1/м);
N = 9: -(Dy^2*Dxx -2*Dx*Dy*Dxy + Dx^2*Dyy)/((Dx^2+Dy^2)*
Sqrt(1+Dx^2+ Dy^2)) - горизонтальная кривизна (1/м);
N = 10: -(Dy^2*Dxx -2*Dx*Dy*Dxy + Dx^2*Dyy)/Sqrt((Dx^2+Dy^2)^3) -
плановая кривизна (1/м);
N=11: -(Dx^2*Dxx+2*Dx*Dy*Dxy+Dy^2*Dyy)/((Dx^2+Dy^2)*
Sqrt((1+Dx^2+Dy^2)^3))- вертикальная (профильная) кривизна (1/м);
N = 12: ((Dx^2-Dy^2)*Dxy-Dx*Dy*(Dxx-Dyy))/Sqrt((Dx^2+Dy^2)^3) - ротор (1/м);
N = 13: D7 – D8 - горизонтальная избыточная кривизна (1/м);
N = 14: D7 + D8 - вертикальная избыточная кривизна (1/м);
N = 15: D6 – D7 - минимальная кривизна (1/м);
N = 16: D6 + D7 - максимальная кривизна (1/м);
N = 17: (Dxx*Dyy – Dxy^2) / ((1. + Dx^2 + Dy^2) * (1. + Dx^2+Dy^2)) - полная
гауссова кривизна (1/м^2).

Для аргументов типа ПОЛИГОН в Калькуляторе реализованы бинарные теоретико-множественные операции: объединение, пересечение, вычитание, которые выполняются с использованием знаков «+», «*», «-», соответственно. Результат выполнения этих операций зависит от анализа взаимного расположения точек пересекающихся и касающихся линий – границ полигонов. Чтобы определить точность представления координат, пользователю рекомендуется перед выполнением операций с полигонами установить единицы измерения линейных размеров аргументов. Для этого в Калькуляторе служит переключатель «м – км».

Функции Калькулятора, принимающие аргументы типов ЛИНИИ или ПОЛИГОН, имеют следующие особенности:

Area(X) - площадь полигона, результат - число. Аргументом также может быть КАРТА, в этом случае вычисляется площадь прямоугольной области грида.

Len(X) - длина линий, результат - число. Возможны также аргументы: ПОЛИГОН, КАРТА (периметр прямоугольника грида), ДАННЫЕ (количество точек).

In_P(X, P) и **Out_P(X, P)** - усечение аргумента X полигоном. Первая из этих функций сохраняет исходный аргумент только внутри полигона, вторая - вне. Для аргумента типа КАРТА размеры и сетка грида не изменяются, лишь значения в «несохраняемых» узлах становятся равными 0. Количество точек в ДАННЫХ в общем случае уменьшается. Лишние ЛИНИИ отбрасываются. Если X – ПОЛИГОН, функция **In_P** осуществляет его пересечение со вторым, а **Out_P** - вычитание второго из первого.

IO_P(X, P) – замена значений аргумента X числом 1, если точка попадает внутрь полигона, иначе – числом 0.

Преимущественно для ДАННЫХ применяются функции:

MIN(X), MAX(X), MIN(X, P), MAX(X, P) – поиск минимума или максимума одного аргумента (возможны все пять типов), для бинарных функций – внутри полигона.

Sum(D), Sum(D, P) – сумма элементов столбца данных – всех точек или только внутри полигона.

Часто используются функции преобразования типов:

C_D(X) - перевод аргумента ЛИНИИ или ПОЛИГОН в ДАННЫЕ. Результирующие точки содержат координаты узлов и атрибуты (значения) линий. Функция непротиворечива аргументу ДАННЫЕ, возвращая его без изменения.

G_P(X) - перевод грида в полигон. Результатом служит прямоугольник, ограничивающий карту. Допускается аргумент ПОЛИГОН, который функцией не изменяется.

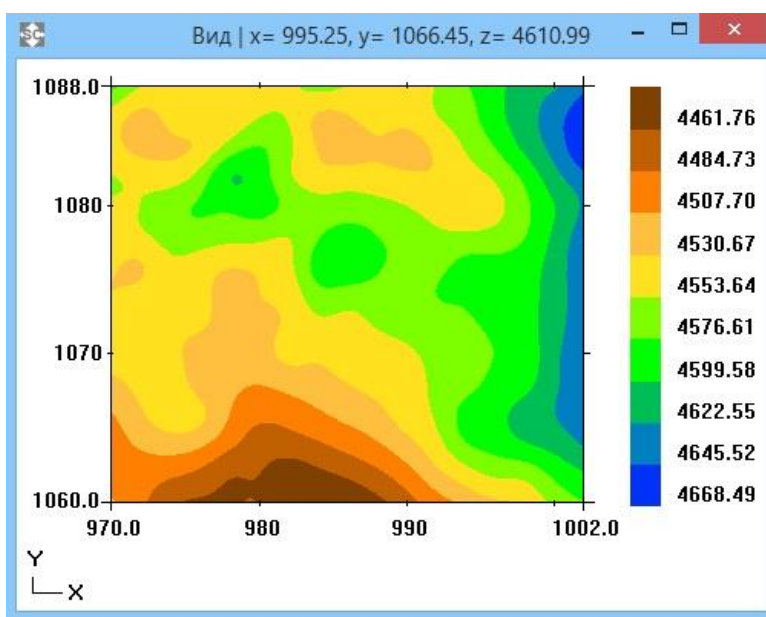
G_D(X) – перевод грида в ДАННЫЕ. Точки результата содержат координаты узлов и значения в узлах грида.

C_G_D(X, G) - снятие с карты значений в заданных точках. Результат - ДАННЫЕ. Допускается один из трех типов первого аргумента: ЛИНИИ, ПОЛИГОН, ДАННЫЕ. Каждый узел линии (точка) координатно накладывается на карту и, в случае попадания в ее границы, получает значение с карты и добавляется к результату. Точки ДАННЫХ, не попадающие на карту, получают отсутствующее значение (outC).

6. Визуализация и сохранение результатов.

В памяти Калькулятора хранится результат вычисления последнего выражения и каждого промежуточного, если таковой был отмечен идентификатором.

Результат любого типа, кроме ЧИСЛА, можно визуализировать в отдельном окне, нажав кнопку «Вид» в главном диалоге. Промежуточный результат, кроме того, доступен для показа из диалога «Значения идентификаторов».



Здесь приведен пример визуализации ГРИДА (рядом показана легенда значений и соответствующих цветов заливки).

Движение курсора в окне визуализации сопровождается отображением координат в заголовке окна. Для ГРИДА, одновременно, выводятся его значения. Для ДАННЫХ выводится значение в точке, которая выбрана левой кнопкой мыши (точка отмечается красным кружком). Следует помнить, что ДАННЫЕ можно визуализировать, если для них заданы столбцы координат.

При визуализации применима функция зуммирования – прямоугольник для увеличения масштаба выделяется левой кнопкой мыши при удержании клавиши Shift. Выход из этого режима – нажатием правой кнопки мыши.

Конечный результат следует сохранять сразу после вычислений, используя кнопку «Сохранить». Для сохранения промежуточного результата его сначала необходимо сделать конечным, то есть ввести в конец табло его идентификатор и нажать кнопку «Вычислить». Другой путь – через диалог «Значения идентификаторов».

Если результат типа КАРТА сохраняется в файле, который инициализирован Калькулятором (для него имеется идентификатор), то дальнейшее обращение к этому файлу будет как к обновленному.

Аналогично – для результата типа ЛИНИИ или ПОЛИГОН.

Сохранение результата типа ДАННЫЕ возможно как в новом файле, так и в файле, загруженном в Калькулятор, то есть представленном на экране окном «Файл данных». Это относится к текстовым файлам формата **dat**. Для формата **shp** потребуются новый файл.

В новом файле создается один (при отсутствии координат) или три столбца в порядке X, Y, Z. В существующем - новый столбец Z добавляется последним, при этом ведется проверка количества точек и совпадения значений их координат (при наличии последних). Изменения в файле тут же отражаются в соответствующем окне «Файл данных».