

При вызове `void Sheet::SetCell(Position pos, std::string text)` Проверяется, что `text` – это формула, если это так, то она создается с помощью `ParseFormula(text.substr(1))` и у нее вызывается `GetReferencedCells()` класса `Formula`, получаем `std::vector<Position> outgoing_edges`. Если он не пуст, вызываем метод класса `Sheet` `IsNoCiclesCreate()`

ОПИСАНИЕ `Sheet::IsNoCiclesCreate(Position pos, const std::vector<Position>& outgoing_edges) const;`

```
{  
  
    Создаем пустой std::set<Position> parents  
  
    Если указатель на ячейку с позицией pos не нулевой, вызываем метод GetAllParent(parent)  
    (Если нулевой, parents остается пустым)  
  
    Добавляем в parents саму позицию pos.  
  
    Создаем std::set<Position> validated_vertices (пустой)  
  
    В цикле проходим все outgoing_edges , для каждой проверяем есть ли она в parent, если  
    есть возвращаем false, если нет, вызываем рекурсивный метод  
  
    Bool HasNoLinkOnParent(validated_vertices, parents );  
  
    Если он вернет false возвращаем false. Если возвращает true, добавляем позицию в  
    validated_vertices.  
  
    Если цикл отработает до конца возвращаем true  
  
}
```

ОПИСАНИЕ `Cell::HasNoLinkOnParent(validated_vertices, parents);`

```
{  
  
    Проходим в цикле все позиции из outgoing_edges_ для каждой проверяем есть ли она в  
    parent, если есть возвращаем false, если нет, вызываем рекурсивный метод  
  
    Bool HasNoLinkOnParent(validated_vertices, parents );  
  
    Если он вернет false возвращаем false. Если возвращает true, добавляем позицию в  
    validated_vertices.  
  
    Если цикл отработает до конца возвращаем true  
  
}
```

Если `IsNoCiclesCreate()` вернул `true` меняем ячейку, если `false` – выбрасываем исключение `CircularDependencyException`