

xquant: PointsList类设计与使用

主要作用

PointsList 被设计成一个类似 **bool** 类型的类。

- 它的定义一般在 `onBar` 外定义成 `static` . `static PointsList<8, 9> MM;`
- 在定义之后, 需要在 `onBar` 中 赋值一个 `bool` 运算。
- `MM = (YB2 && (X5 > X5.ref(1) || X4 > X4.ref(1)));` 这个运算是可以在切片中运行的。
- 它可以和同类进行bool运算。 `MM = MM && m_line;`
- 它有一个 **bool** 类型的成员, 记录切片运算的当前 `bool` 值。 `MM.b_ = true;`
- 它有一个 **vector** 成员: `MM.points_`, 记录所有当 `MM.b_` 为真时的 `OHLCV` 数组 `index` 值。
- 它还可以进行 `bool` 判别: `if (MM == true) {MM = NN;}` `if(MM) {}`
- 不同类型的比较 `if(MM.value_ == NN.value_){}`

```
template <int... N> class PointsList : public BoolBase<N...>
template <int... N> class BoolBase : public SectionsBase<N...>
template <int... N> class SectionsBase : public PeriodBase<N...>
template <int... N> class BoolSerial : public BoolBase<N...>
template <int... N> class SegmentsList : public BoolSerial<N...>
template <int... N> class FutureSerial : public SectionsBase<N...>
template <int... N> class Future : public PeriodBase<N...>
template <int... N> class MA_t : public FutureSerial<N...>, public OhlcvSerial<N...>
template <int... N> class EMA_t : public FutureSerial<N...>
```

主要作用

用于记录当某种情况发出时的位置，这个位置是一个int型的 数组的下标值。多个这样的值保存一个 `vector<int>` 中。

比如，当5日线上穿10日线的那一天，`value_`将变为true，同时会记录这一天的index值。

```
1     public:
2         std::vector<int> points_;
3         bool value_;
```

```
1     for (auto ix : IC0132_B.points_) {
2         LOG(INFO) << ix << " O: " << IC0132_B.O[ix] << " H: " << IC0132_B.H
[ix] << " L: " << IC0132_B.L[ix] << " C: " << IC0132_B.C[ix];
3     }
```

```

1  // test -----begin
2  static PointsList<8, 9> MM;
3  static PointsList<8, 9> NN;
4
5  static PointsList<8, 12> MT12;
6  static PointsList<8, 12> NT12;
7
8  MM = m_line;
9  bool test01 = MM && m_line;
10 bool test02 = m_line && MM;
11
12 MM = MM;
13 MM = NN;
14
15 MM = MT12 && true;
16
17 MM = MM && m_line;
18 MM = m_line && MM;
19 MM = !m_line;
20 MM = !MM;
21
22 test01 = MM && NN;
23 test01 = MM || NN || m_line;
24
25 ▾ if (MM != true) {
26     MM = NN;
27 }
28
29 ▾ if (MM == true) {
30     MM = NN;
31 }
32 ▾ if (true == MM) {
33     MM = NN;
34 }
35 ▾ if (true != MM) {
36     MM = NN;
37 }
38
39 ▾ if (NN == MM) {
40     MM = NN;
41 }
42 ▾ if (NN != MM) {
43     MM = NN;
44 }
45
46 // test -----end

```

