EMT

Как построить полную модель с использованием Excel и Z3

Buy3 Buy2 Buy1 Sell1 Sell2 Sell3

Price

Price

```
ystem;
ystem.Collections.Generic;
ystem.Linq;
ystem.Text;
ystem.Threading.Tasks;

ouse

ing name;
ing homeNumber;
dent[] students;
```

Buy3 Buy2 Buy1 Sell1 Sell2 Sell3

Nickname/eMail

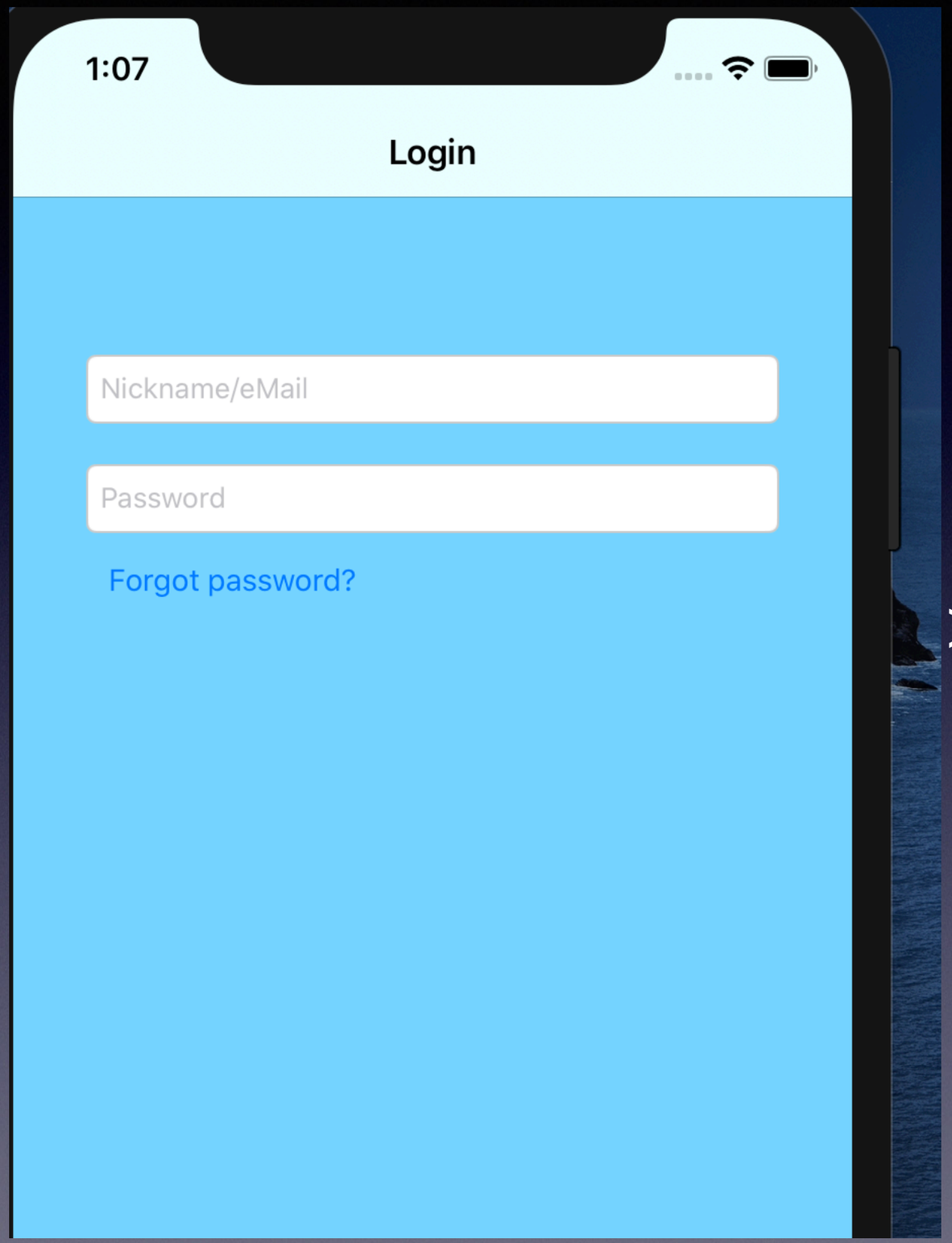
Password

Forgot password?

```
g.Tasks;
```

Пр гы

```
s;
```


1:07
Login
Nickname/eMail
Password
Forgot password?

Генерация случайных тестов

- ***GetHyperState(model)***
- ***IsPossible(model, input)***
- ***GetStartPosition()***
- ***StopCondition(model)***
- ***GetInfo()***

Генерация случайных тестов

```
1.  info = GetInfo()
2.  model = GetStartPosition()
3.  base = {}
4.  for count = 1..n:
5.      do
6.          input = rand(info.input)
7.      while not IsPossible(model, input)
8.      newmodel = Next(model, input)
9.      base.write(GetHyperState(model) -> GetHyperState(newmodel))
10.     model = newmodel
11.     while StopCondition(model):
12.         model = GetStartPosition()
```

Дополнение полученного графа ненайденными вершинами

```
1.  queue = Queue(base.keys())
2.  while queue not empty:
3.      model = queue.first
4.      newmodel = None
5.      input = None
6.      if z3.find(And(
7.                  IsPossible(model, input),
8.                  Next(model, input) == newmodel,
9.                  GetHyperState(newmodel) not in base)):
10.         base.write(model -> newmodel)
11.         Continue
12.     else:
13.         queue.pop()
```

Дополнение полученного графа ненайденными ребрами

```
1.  queue = Queue(base.keys())
2.  while queue not empty:
3.      model = queue.first
4.      newmodel = None
5.      input = None
6.      if z3.find(And(
7.                  IsPossible(model, input),
8.                  Next(model, input) == newmodel,
9.                  GetHyperState(newmodel) not in base[model])):
10.         base.write(model -> newmodel)
11.         Continue
12.     else:
13.         queue.pop()
```

Рассмотрим пример

Реализация базовых функций

```
1. Function HSymb(x)
2.     If x = 1 Then
3.         HSymb = «1»
4.     ElseIf x = 0 Then
5.         HSymb = «0»
6.     Else
7.         HSymb = «M»
8.     End If
9. End Function
10.
11.Function GetHyperState(b2, b1, a1, a2)
12.    GetHyperState = HSymb(b2) & HSymb(b1) & «|» & HSymb(a1) & HSymb(a2)
13.End Function
```

Реализация базовых функций

```
1. Function IsPossible(b2, b1, a1, a2, type, price, volume)
2.     If b2 <> 0 And type = «Buy» Or a2 <> 0 And type = «Sell» Then
3.         IsPossible = False
4.     Else
5.         If volume < 1 Or b2 < 0 Or b1 < 0 Or a1 < 0 Or a2 < 0 Then
6.             IsPossible = False
7.         Else
8.             If type = «Buy» Or type = «Sell» Then
9.                 IsPossible = True
10.             Else
11.                 IsPossible = False
12.             End If
13.         End If
14.     End If
15.End Function
```

Реализация базовых функций

```
1. Function GetStartPosition()
2.     GetStartPosition = Array(0, 0, 0, 0)
3. End Function
```

Реализация базовых функций

```
1. Function StopCondition(b2, b1, a1, a2)
2.     StopCondition = False
3. End Function
```

Реализация базовых функций

```
1. Function GetInfo()
2.     GetInfo = Array(4, 3, Array(«Buy», «Sell»), Array(101), Array(1, 2, 3, 4,
   5, 6, 7, 8, 9))
3. End Function
```

Результат работы 100 случайных тестов

00|00: {'00|M0'}

00|M0: {'00|00', '00|MM'}

00|MM: {'00|10', '00|M0', '00|MM'}

00|10: {'00|1M'}

00|1M: {'0M|00'}

0M|00: {'MM|00', '0M|00', '00|M0'}

MM|00: {'M1|00', '0M|00'}

M1|00: {'01|00', '0M|00'}

01|00: {'M1|00'}

9 вершин; 16 ребер

Результат работы верификатора

00|00: {'**00I10**', '**01I00**', '**0MI00**', '00|M0'}

00|M0: {'**0MI00**', '00|MM', '**00IM1**', '00|00', '**00I10**', '**01I00**', '**00IM0**'}

00|MM: {'**0MI00**', '00|MM', '**00I00**', '00|10', '**01I00**', '**00I1M**', '00|M0'}

00|10: {'**0MI00**', '**00I11**', '**00I00**', '**01I00**', '00|1M'}

00|1M: {'0M|00', '**00I00**', '**00I10**', '**01I00**', '**00IM0**'}

0M|00: {'MM|00', '**1MI00**', '0M|00', '**00I00**', '**00I10**', '**01I00**', '00|M0'}

MM|00: {'M1|00', '**MMI00**', '0M|00', '**00I00**', '**00I10**', '**01I00**', '00|M0'}

M1|00: {'0M|00', '**00I00**', '**00I10**', '**01I00**', '00|M0'}

01|00: {'M1|00', '**00I00**', '**00I10**', '**11I00**', '**00IM0**'}

00IM1: {'**0MI00**', '**00IM1**', '**00I11**', '**00I00**', '**00I10**', '**01I00**'}

00I11: {'**00I00**', '**00I10**', '**01I00**', '**0MI00**'}

1MI00: {'**1MI00**', '**00I00**', '**00I10**', '**01I00**', '**11I00**', '**00IM0**'}

11I00: {'**00I00**', '**00I10**', '**01I00**', '**00IM0**'}

13 вершин; 72 ребра