Sistemas Distribuídos

1^a Frequência

	27 de Outubro de 2010	Duração: 1h	
Nome:			Número:
	I		
	do "Jantar de Filósofos" apresentado nas a a, por erro, ficam num ciclo while(1);. Que		
2 Qual a necessidade e/corrência?	ou vantagem de envolver o sistema operat	ivo na concretização de prin	nitivas de controlo de con

II

Corrija o código Java apresentado abaixo e procure maximizar o seu paralelismo potencial.

```
import java.util.ArrayList;
import java.util.Random;

class Weight {

float value = 0;

public Weight(float value) { this.value = value; }

public float get() { return value; }

private void set(float value) { this.value = value; }

public Weight split() {
 value = value / 2;
 return new Weight(value);
}
```

```
19
          public void merge(Weight other) {
20
21
              value = value + other.get();
other.set(0);
22
23
    }
24
25
     class ConstantWeightSet {
26
27
          ArrayList<Weight> weights = new ArrayList<Weight>();
28
29
         public ConstantWeightSet(int value) {
30
              weights.add(new Weight(value));
31
32
33
         public int size() { return weights.size(); }
34
35
         public float get(int i) {
36
              return weights.get(i).get();
37
38
         public void split(int i) {
    Weight splitweight = weights.get(i).split();
39
40
41
              weights.add(splitweight);
         }
43
44
          public\ void\ \texttt{merge}(int\ i\ ,\ int\ j\ )\ \big\{
45
              weights.get(i).merge(weights.get(j));\\
46
              weights.remove(j);
47
48
49
          public float total() {
              float sum = 0;
for (int i = 0; i != weights.size(); i++)
50
51
              sum += weights.get(i).get();
return sum;
52
53
54
         }
55
    }
57
     class RandomRun extends Thread {
58
59
         final int M = 100;
60
         ConstantWeightSet weights;
61
62
          public RandomRun(ConstantWeightSet weights) {
63
              this.weights = weights;
64
65
         public void run() {
66
              Random gen = new Random();
67
              for (int i = 0; i != M; i++) {
    int op = (weights.size() == 1) ? 0 : gen.nextInt(3);
    int a = gen.nextInt(weights.size()), b;
69
70
71
72
73
74
                   switch \ (\texttt{op}) \ \{
                       case 0:
                             weights.split(a);
                            break;
75
76
77
                            do b = gen.nextInt(weights.size()); while (a == b);
                             weights.merge(a\,,\ b\,);
78
                            break:
79
                        case 2:
                            System.out.printf("\%g\n", weights.total());
81
                   }
82
             }
         }
83
    }
84
85
86
     class Main {
87
         public static void main(String[] args) {
88
              ConstantWeightSet weights = new ConstantWeightSet(4);
              89
90
91
                   rr.start();
93
              }
94
         }
    }
95
```