Heimadæmi – heimadæmi 13 hópverkefni

Arnar Sigurðsson

public class V13

{

  // Hvert tilvik af klasanum Container er ílát fyrir

  // eina heiltölu. Margir þræðir geta samtímis unnið

  // með ílátið.

  static class Container

  {

    boolean isEmpty = true;

    int theValue;

    // Fastayrðing gagna:

    //   Ílátið er tómt þþaa isEmpty sé true.

    //   Ef ílátið er ekki tómt þá inniheldur

    //   theValue gildið í ílátinu.

    // Notkun: c.put(x);

    // Eftir:  Búið er að setja x í ílátið c.

    //         Ef til vill þurfti að bíða eftir

    //         að ílátið tæmdist áður en það

    //         tókst.

    public synchronized void put( int x )

      throws InterruptedException

    {

      while( !isEmpty ) wait();

      isEmpty = false;

      theValue = x;

      notifyAll();

    }

    // Notkun: x = c.get();

    // Eftir:  Búið er að sækja x úr ílátinu c.

    //         Ef til vill þurfti að bíða eftir

    //         að ílátið fylltist áður en það

    //         tókst.

    public synchronized int get()

      throws InterruptedException

    {

      while( isEmpty ) wait();

      int x = theValue;

      isEmpty = true;

      notifyAll();

      return x;

    }

  }

  // Tilvik af klasanum Producer eru þræðir sem senda tölurnar

  // 1,2,...,10000 í ílát (Container) sem þeir hafa tilvísun í

  // og deyja svo drottni sínum.

  static class Producer extends Thread

  {

    final Container c;

    // Fastayrðing gagna:

    //   c er ílátið sem þessi Producer sendir tölur í.

    // Notkun: Producer prod = new Producer(c);

    // Fyrir:  c er Container.

    // Eftir:  prod er Producer sem setur tölur í c.

    public Producer( Container c )

    {

      this.c = c;

    }

    // Notkun: prod.run();

    // Fyrir:  Ekkert.

    // Eftir:  prod er búinn að senda 1 til 10000 í sinn

    //         Container.

    public void run()

    {

      try

      {

        for(int i=1; i<10001; i++){

          // Fjöldi gilda sem hafa verið sett í containerinn c

          // er á bilinu 0 - 10000.

          c.put(i);

      }

      }

      catch( InterruptedException e )

      {

    e.printStackTrace();

      }

    }

  }

  // Tilvik af klasanum Consumer eru þræðir sem taka

  // 10000 tölur úr íláti (Container) sem þeir

  // hafa tilvísun í, skrifa svo summu talnanna, og

  // deyja svo drottni sínum.

  static class Consumer extends Thread

  {

    final Container c;

    // Fastayrðing gagna:

    //   c er ílátið sem þessi Consumer sækir tölur úr.

    // Notkun: Consumer cons = new Consumer(c);

    // Fyrir:  c er Container.

    // Eftir:  cons er Consumer sem sækir tölur úr c.

    public Consumer( Container c )

    {

      this.c = c;

    }

    // Notkun: cons.run();

    // Fyrir:  Ekkert.

    // Eftir:  cons er búinn að sækja 10000 tölur

    //         úr sínum Container og skrifa summuna.

    public void run()

    {

      try

      {

        long sum = 0;

      for(int i = 1; i<10001; i++){

        // Fjöldi gilda sem hafa verið tekin úr container c

        // er á bilinu 0 - 10000.

        sum = sum + c.get();

      }

        System.out.println(sum);

      }

      catch( InterruptedException e )

      {

    e.printStackTrace();

      }

    }

  }

  // Notkun: (á skipanalínu) java V13

  // Fyrir:  Ekkert.

  // Eftir:  Búið er að skrifa tvær tölur sem samanlagt

  //         eru tvöfalt 1+2+...+10000

  public static void main( String[] args )

  {

    Container c = new Container();

    Producer prod1 = new Producer(c);

    Producer prod2 = new Producer(c);

    Consumer cons1 = new Consumer(c);

    Consumer cons2 = new Consumer(c);

    prod1.start();

    cons1.start();

    prod2.start();

    cons2.start();

  }

}