Data Structures & Algorithms

Subodh Kumar

(subodh@iitd.ac.in, Bharti 422)

Dept of Computer Sc. & Engg.

Exceptions are your friend

- Helps you handle run-time errors
  - And think about types of errors that are possible
- Method raises exception to indicate "unexpected" condition
  - Implicit "return flags"
- Catch Expected exception and handle it
  - Separates error handling in a modular way
  - Eases "passing the buck" to caller
- Unhandled exception terminates the program
  - Program should handle user defined exceptions

Unchecked Exception

```
public  class  Propagate {
   void  divide() {
      int  m = 25,  i = 0;
      i = m / i;
   }
   void  process() {
      divide();
   }
   public static void main(String[] args) {
      Propagate p = new Propagate();
      p.process();
   }
}
```

Exception

```
java.lang.ArithmeticException:     / by zero
        at Propagate.orange(Propagate.java:4)
        at Propagate.apple(Propagate.java:8)
        at Propagate.main(Propagate.java:11)
```

JAVA default handler

Exception Handling

```
try {
    … normal program code
}

catch(Exception e) {
    … exception handling code
}

finally { // optional: execute after try
}
```

Exception Handling

```
try {
    … normal program code
}

cacatch (Exception e) {                    xception e) {
    … exception handling code
}

catch(SomeExceptionClass e) {
    … exception handling code
}

finally { // this is optional
}
```

System Exception

```
public void aMethod() {
   try {
      int a[] = new int[2];
      a[2] = 1;
  } catch (ArrayIndexOutOfBoundsException e)
  {
      System.out.println(
             "exception: " + e.getMessage());
      e.printStackTrace();
  }
}
```

Pass Exception Along

- A method that might encounter an unhandled exception, should use “throws” clause:

```
public void myMethod throws IOException
{
     … normal code with some I/O
}
```

- It can generate exception as well

Example

```
class MyException extends Exception  {}

class MyClass {
   void someMethod() throws Myexception {
        MyException x = new MyException();
        // ... some code here
        if (val < 1) throw x;
        }
}
```

Example

```
class MyException extends Exception  {
   MyException(String s) { super(s); }
}
…

void someMethod throws MyException {
   MyException x = new MyException(“Message”);
   ...
   if (val < 1) throw x;
}
```