In C, things are far from straightforward

errors_c.c

```c
#include <stdio.h>
#include <errno.h>

int main(void) {
    int s;

    s = socket(...);
    if (s < 0) {
        fprintf(stderr, "socket() failed: %s\n", strerror(errno));
        exit(1);
    }
}
```

… and remember your experiences with using scanf with stdin from the terminal.

Catching Exceptions

- sometimes, things fail, for example: parsing
- when this happens, this is called an `exceptional circumstance' or `exception', which should not in all cases lead to the program exiting
- a try-catch block can handle exceptions without the program breaking:

```
try {
  // do the things that may go wrong here
} catch (Exception e) {
  // do the things that should happen when
  // something went wrong here,
  // the object e provides information about what happened
}
```

An Example of Handling Exceptions

Adder.java

```
class Adder {

    int sum;

    Adder() {
        sum = 0;
    }

    void add(int summand) {  sum
        += summand;
    }
}
```

```
$ java Calculator 3 5 two
Something went wrong, but I can handle it!

$
```

code does not break anymore and exits normally

ExceptionalCalculator.java

```
class ExceptionalCalculator {

  public static void main (String[] args) {
    Adder adder = new Adder();
    try {
      for (String arg : args) {
          adder.add(Integer.parseInt(arg));
      }
      System.out.println("Sum:" + adder.sum);
    } catch (Exception e) {
      System.out.println("Something went wrong, but I can handle it!");
} } }
```

this code block is secured – if an exception happens (e.g. parsing fails) , the program does not stop, but jumps to the start of the catch block

An (Better) Example of Handling Exceptions

Adder.java

```
class Adder {

    int sum;

    Adder() {
        sum = 0;
    }

    void add(int summand) {
        sum += summand;
    }
}
```

ExceptionalCalculator2.java

```
public class ExceptionalCalculator2 {

  public static void main (String[] args) {
    Adder adder = new Adder();
    try {
      for (String arg : args) {
          adder.add(Integer.parseInt(arg));
      }
      System.out.println("Sum:" + adder.sum);
    }
   catch (NumberFormatException e) {
      System.out.println(e.getMessage());
        System.out.println("Was that really an integer?");
      //e.printStackTrace();
   }
   catch (Exception e) {
      System.out.println(e.getMessage());
      System.out.println("Something went wrong");
   }
} }
```