Lack of Rigorous Error Check (LREC)

Example: THPSize_init is a module in PyTorch related to initialization of the tensor size. Listing 1 illustrates its code structure in version 0.3.1 [1]. The PyModule_AddObject() function adds an object named Size to this module, but this process lacks rigorous error checking. In PyTorch version 1.1.0 [2], the module underwent refactoring, as shown in Listing 2, which includes a check on the return value of the PyModule_AddObject() function.

```
1  bool THPSize_init(PyObject *module)
2  {
3    THPSizeClass = (PyObject*)&THPSizeType;
4    if (PyType_Ready(&THPSizeType) < 0)
5      return false;
6    Py_INCREF(&THPSizeType);
7    PyModule_AddObject(module, "Size", (PyObject *)&THPSizeType);
8    return true;
9  }
```

Listing 1. Design Smell - Lack of Rigorous Error Check

```
1  void THPSize_init(PyObject *module)
2  {
3    if (PyType_Ready(&THPSizeType) < 0) {
4      throw python_error();
5    }
6    Py_INCREF(&THPSizeType);
7    if (PyModule_AddObject(module, "Size", (PyObject*)&THPSizeType) < 0) {
8      throw python_error();
9    }
10 }
```

Listing 2. Refactoring - Lack of Rigorous Error Check

---

[1] https://github.com/pytorch/pytorch/blob/v0.3.1/torch/csrc/Size.cpp[138-146]
[2] https://github.com/pytorch/pytorch/blob/v1.1.0/torch/csrc/Size.cpp[198-207]