



universidade de aveiro
escola superior de tecnologia e
gestão de águeda

Introdução à Programação

Exceções

Mesmo que uma instrução ou expressão esteja sintaticamente correta, ela pode causar um erro ao tentar executá-la.

Erros detetados durante a execução são chamados de exceções e não são incondicionalmente fatais.

A maioria das exceções não são tratadas por programas e resultam em mensagens de erro.

Exceções

```
>>>10 * (1/0)
```

ZeroDivisionError: division by zero

```
>>> 4 + spam*3
```

NameError: name 'spam' is not defined

```
>>> '2' + 2
```

TypeError: can only concatenate str (not "int") to str

A string impressa como o tipo de exceção é o nome da exceção interna que ocorreu.

Os tipos de exceção do exemplo são: ZeroDivisionError, NameError e TypeError.

O restante da linha fornece detalhes com base no tipo de exceção e no que a causou.

Handling Exceptions

É possível escrever programas que lidam com exceções seleccionadas.

Observemos o exemplo seguinte que solicita a entrada de um inteiro por parte do utilizador.

```
>>>while True:
    try:
        x = int(input("Please enter a number: "))
        break
    except ValueError:
        print("Oops! That was no valid number. Try again...")
```

Handling Exceptions

A instrução ***try*** funciona da seguinte maneira.

Primeiro, a cláusula ***try*** é executada.

Se nenhuma exceção ocorrer, a cláusula ***except*** será ignorada e a execução da instrução ***try*** será encerrada.

Se ocorrer uma exceção durante a execução da cláusula ***try***, o restante da cláusula será ignorado.

Então, se o seu tipo corresponder à exceção nomeada após a palavra-chave ***except***, a cláusula ***except*** é executada e a execução continua após o bloco ***try/except***.

Se ocorrer uma exceção que não corresponda à exceção nomeada na cláusula ***except***, ela será passada para as instruções ***try*** externas. Se nenhum manipulador for encontrado, será uma exceção não tratada e a execução será interrompida com uma mensagem conforme mostrado anteriormente.

Handling Exceptions

Uma instrução *try* pode ter mais de uma cláusula *except*, para especificar manipuladores para diferentes exceções.

No máximo um manipulador será executado. Os manipuladores lidam apenas com exceções que ocorrem na cláusula *try* correspondente e não noutros manipuladores da mesma instrução *try*.

Uma cláusula *except* pode nomear várias exceções como um tuplo entre parênteses, por exemplo:

```
except (RuntimeError, TypeError, NameError):  
    pass
```

Raising Exceptions

A instrução raise permite ao programador forçar a ocorrência de uma exceção específica.

```
>>>raise NameError('HiThere')
```

```
NameError: HiThere
```

O único argumento indica a exceção a ser lançada.

Deve ser uma instância de uma exceção ou uma classe de exceção.