Introduction to Programming

Functions

- Simple Decision
- Two-Way Decision
- **Multi-Way Decision**
- **Exception Handling**
- **Study in Design: Max of Three**

Multi-Way Decision

- Make a three way-decision
- Convert a students numerical grade to a letter grade as follows:
  - If numerical grade is above 80 then letter grade is A
  - If numerical grade is between 60 and 79 the letter grade is B
  - If numerical grade is less then 60 then letter grade is C.
  - Easy Way
- Three if statements!

Multi-Way Decision

- We can also do this using two if-else statements
  - One inside the other
  - Putting if statement inside another is called nesting

Multi-Way Decision

- Let expend this requirements as follows:

Numerical Score	Letter Grade
100 to 90	A
89 to 80	B
79 to 70	C
69 to 60	D
0 to 59	F

- It will be hard to keep track of nesting
- We can use `elif` construct

Multi-Way Decision

```
if <condition1>:
   <case1 statements>
elif <condition2>:
   <case2 statements>
elif <condition3>:
   <case3 statements>
…
else:
   <default statements>
```

Multi-Way Decision

- This form sets off any number of mutually exclusive code blocks.
- Python evaluates each condition in turn looking for the first one that is true. If a true condition is found, the statements indented under that condition are executed, and control passes to the next statement after the entire `if-elif-else`.
- If none are true, the statements under `else` are performed.
- The `else` is optional. If there is no `else`, it's possible no indented block would be executed.

Error Handling

- What happen if the user enters an incorrect type of data?
- We can use isnumeric or isdigit methods
  - They belong to string class

Exception Handling

- In the last example we use decision structure to avoid converting non-numerical string into int
- This is true for many programs: decision structures are used to protect against rare but possible errors.
- Sometimes programs get so many checks for special cases that the algorithm becomes hard to follow.
- Programming language designers have come up with a mechanism to handle exception handling to solve this design problem.

Exception Handling

- The programmer can write code that catches and deals with errors that arise while the program is running, i.e., “Do these steps, and if any problem crops up, handle it this way.”
- This approach obviates the need to do explicit checking at each step in the algorithm.

Exception Handling

- The try statement has the following form:

```
try:
     <body>
except <ErrorType>:
     <handler>
```

- When Python encounters a try statement, it attempts to execute the statements inside the body.
- If there is no error, control passes to the next statement after the `try...except`.

Exception Handling

- If an error occurs while executing the body, Python looks for an except clause with a matching error type. If one is found, the handler code is executed.
- How do we know error types?
- Example

Some Lessons

- There's usually more than one way to solve a problem.
  - Don't rush to code the first idea that pops out of your head. Think about the design and ask if there's a better way to approach the problem.
  - Your first task is to find a correct algorithm. After that, strive for clarity, simplicity, efficiency, scalability, and elegance.

Some Lessons

- Be the computer.
  - One of the best ways to formulate an algorithm is to ask yourself how you would solve the problem.
  - This straightforward approach is often simple, clear, and efficient enough.

Some Lessons

- Generality is good.
  - Consideration of a more general problem can lead to a better solution for some special case.
- Don't reinvent the wheel.
  - If the problem you're trying to solve is one that lots of other people have encountered, find out if there's already a solution for it!
  - As you learn to program, designing programs from scratch is a great experience!
  - Truly expert programmers know when to borrow.
  - Don't forget to give credit!

Some Lessons

- If it does not work
  - Then try to explain it to someone (non-CS person)
  - If no one is willing to listen the tell it to rubber ducky!