Introduction to Programming

Spring 2022

Objectives

- Software development process
- Application of software development process
- Elements of a program
- Keeping your programs organized

Software Development Process

- The process of creating a program is divided into stages
  - Each stage will produce different set of information

Software Development Process

- Analyze the Problem
  - Figure out exactly the problem to be solved.
  - Try to understand it as much as possible.

Software Development Process

- Determine Specifications
  - Describe exactly what your program will do.
  - Don’t worry about how the program will work, but what it will do.
  - Includes describing the inputs, outputs, and how they relate to one another.

Software Development Process

- Create a Design
  - Formulate the overall structure of the program.
  - This is where the how of the program gets worked out.
  - Develop your own algorithm that meets the specifications.

Software Development Process

- Implement the Design
  - Translate the design into a computer language.
  - In this course we will use Python.

Software Development Process

- Test/Debug the Program
  - Try out your program to see if it worked.
  - If there are any errors (bugs), they need to be located and fixed. This process is called debugging.
  - Your goal is to find errors, so try everything that might “break” your program!

Software Development Process

- Maintain the Program
  - Continue developing the program in response to the needs of your users.
  - In the real world, most programs are never completely finished – they evolve over time.

Software Development Process

- Analysis
  - The temperature is given in Celsius, user wants it expressed in degrees Fahrenheit.
- Specification
  - Input – temperature in Celsius
  - Output – temperature in Fahrenheit
  - Output = 9/5(input) + 32

Software Development Process

- Design
  - Input, Process, Output (IPO)
  - Input → Prompt the user for input (Celsius temperature)
  - Process → Convert input to Fahrenheit using F = 9/5(C) + 32 formula
  - Output → Display the result by displaying it on the screen

Software Development Process

- Pseudocode

```
1) Display "What is the Celsius temperature?"
2) Input degrees Celsius (call it celsius)
3) Calculate degree Fahrenheit using the formula (9/5)*celsius+32 (call this fahrenheit)
4) Display the value in fahrenheit.
```

Software Development Process

```
# convert.py
# A program to convert Celsius temps to
# Fahrenheit

def main():
    celsius = eval(input("What is the
          Celsius temperature? "))
    fahrenheit = (9/5) * celsius + 32
    print("The temperature is ",
       fahrenheit," degrees Fahrenheit.")
```

Software Development Process

- Once we write a program, we should test it!
- Test Data

Degree Celsius	Degree Fahrenheit
0	32
100	212
-40	-40

Elements of Programs

- main()
  - This is the name of our function
  - Python programs can contain multiple functions
  - Each function will start with key word “def” and will include the name of the function
- More about functions later in the course
  - By convention function that will start the execution of the program is called main

Elements of Programs

- All text after # will be ignored by Python
- # are used to add comments to your program
- Comments provide information to user and programmer
- Always start with a block of comments at top which describes the program and list authors
  - You can also list version information
- Can be used to remove part of the program without deleting it.

Elements of Programs

- Names
  - Names are given to variables (celsius, fahrenheit), modules (main, convert), etc.
  - These names are called identifiers
  - Every identifier must begin with a letter or underscore (“_”), followed by any sequence of letters, digits, or underscores.
  - Identifiers are case sensitive.
  - Cannot use keywords as identifiers

Elements of Programs

- Names
  - Cannot use keywords as identifiers
  - Example: False, class, None
  - Can use name of built-in functions as identifiers
- It is generally a very bad idea to do so!
  - Example: print, eval

Elements of Programs

- These are all different, valid names
  - X
  - Celsius
  - FirstName
  - last_name
  - address01

Elements of Programs

- Expressions
  - The fragments of code that produce or calculate new data values are called expressions.
  - Literals are used to represent a specific value, e.g. 3.9, 1, 1.0
  - Simple identifiers can also be expressions.
  - Also included are strings (textual data) and string literals (like "Hello").

Elements of Programs

- Simpler expressions can be combined using operators.
- + Addition
- - Subtraction
- * Multiplication
- / Division
- % Modulus
- ** Exponentiation
- // Floor division

Elements of Programs

- Spaces are irrelevant within an expression.
- The normal mathematical precedence applies.
- 4 + 5
- 4 + 5 * 3
- (4 + 5) * 3
- “Jack “ + “Doe”

Class Work

- The Mars Curiosity rover takes photos of the Mars surface, and then transmits them to NASA at the speed of light. Light travels at about 186,000 miles per second. Write a program to calculate how long it takes a photo from Curiosity to reach NASA when Mars is at its closest orbit to Earth, a distance of about 34 million miles.