Hacettepe University

Computer Engineering Department

Programming in python

BBM103 Introduction to Programming Lab 1
Week 11

Fall 2018

Debugging

- ***Debugging*** is the process of identifying and removing errors that prevent correct operation of computer software or a system.

- PyCharm provides a full range of facilities for debugging your source code:
  - Breakpoints in Python.
  - Customizable breakpoint properties: conditions, pass count, etc.
  - Frames, variables, and watches views in the debugger UI.
  - Runtime evaluation of expressions.

- For detailed explanation of the debugging process in PyCharm:
  https://www.jetbrains.com/help/pycharm/2016.1/debugging.html

Debugging Cont.

- **General debugging steps:**

  1. Configure the debugger options.
  2. Define a run/debug configuration.
  3. Create breakpoints in the source code.
  4. Launch a debugging session.
  5. Pause or resume the debugging session as required.
  6. During the debugger session, step through the breakpoints, evaluate expressions, change values on-the-fly , examine suspended program, explore frames, and set watches .

Starting the Debugger Session

- Open the desired Python script in the editor, or select it in the Project tool window.

- On the context menu, choose Debug <script name>:

- Set breakpoints in the source code.
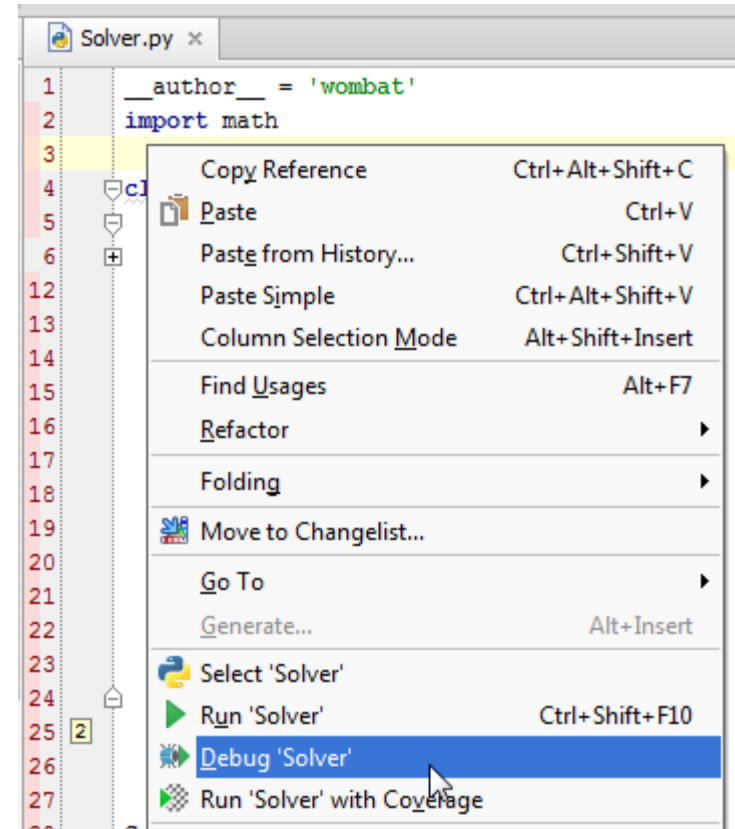
PyCharm Debug Tool Window

- The Debug tool window becomes available when you start debugging.

- It displays the output generated by the debugging session for your application.

- For Toolbars and Items descriptions:
  https://www.jetbrains.com/help/pycharm/2016.1/debug-tool-window.html

Lab Exercise - 1

- Could you write a function that tells us that "120" is "5!"?
- **Hint:** The strategy is pretty straightforward, just divide the term by successively larger terms until you get to "1" as the result: (use recursion)

```
120 -> 120/2 -> 60/3 -> 20/4 -> 5/5 -> 1 => 5!
```

- **python3.5 myExercise1.py 120,150**

```
Output

120 = 5!
150 NONE
```

Use recursion

Lab Exercise - 2

- An anagram is a form of word play, where you take a word (or set of words) and form a different word (or different set of words) that use the same letters, just rearranged.

- **Hint :** All words must be valid spelling, and shuffling words around doesn't count.

- Write a function that takes an input and checks if two words in each line are anagrams.

input.txt

```
Clint Eastwood ? Old West Action
parliament ? partial man
```

python3.5 myExercise2.py input.txt

```
Output:
Clint Eastwood is an anagram of Old West Action
parliament is NOT an anagram of partial man
```