CS 150 - Midterm “Cheat Sheet”

Input/Output

- Reading input from the user

  `input(message)`: Displays *message* to the user and returns what the user typed as a string

- Reading from a file

  ```
  file = open(filename, "r")
  for line in file:
      # do something with line (a string)
  file.close()
  ```

Strings

- The following functions are built-in and answer questions about strings

  `len(string)`: Returns the number of characters in the string

  `int(string), float(string)`: Converts a string to an int or float

- String object methods

  `upper(), lower(), capitalize()`: Returns a new string that is upper or lower-cased, or capitalized

  `find(some_string)`: Returns the index that *some_string* occurs at in the string or -1 if it does not occur

  `find(some_string, index)`: Same as above, but starts searching at *index*

  `replace(old, new)`: Return a copy of the string with all occurrences of *old* substituted with *new*

  `startswith(prefix)`: Returns True if the string starts with *prefix*, False otherwise

  `endswith(suffix)`: Returns True if the string ends with *suffix*, False otherwise

  `strip()`: Returns a copy of the string with leading and trailing whitespace removed

  `split()`: Return a list of the words in the string using a space as the delimiter

- String operators

  `string1 + string2`: Returns a new string that is the concatenation of *string1* and *string2*

  `string * int`: Returns a new string that is *string* repeated *int* times

Lists

- The following functions are built-in and answer questions about lists

  `len(list)`: Returns the number of elements in *list*

  `sum(list), min(list), max(list)`: Returns the sum, min, or max of elements in *list*

- List object methods

  `append(x)`: Adds *x* to the end of the list

  `insert(index, x)`: Insert *x* at *index* in the list

  `pop()`: Removes the item at the end of the list and returns it

  `pop(index)`: Removes item at *index* from the list and returns it

  `reverse()`: Reverses the elements in the list

  `sort()`: sorts the elements in the list

- List operators

  `list1 + list2`: Returns a new list that contains the elements of *list1* followed by the elements of *list2*

  `list * int`: Returns a new list that contains the items in *list* repeated *int* times

Modules

- `turtle` module

  `forward(distance)`: Move the turtle forward by the specified *distance*

  `right(angle) left(angle)`: Turn the turtle right/left by *angle*

  `goto(x, y)`: Move turtle to position *x, y*

  `setheading(angle)`: Set the turtles heading to angle

  `circle(radius)`: Draw a circle with specified *radius*; the center is *radius* units left of the turtle

  `penup()`: Pull the pen up – no drawing when moving

  `pendown()`: Put the pen down – drawing when moving

  `fillcolor(color)`: Change the fill color to *color*, where *color* is a string

  `begin_fill()`: Start filling

  `end_fill()`: Fill in the shape drawn since the last call to `begin_fill`

- `random` module

  `randint(a, b)`: Return a random integer *N* such that *a ≤ N ≤ b*

  `uniform(a, b)`: Return a random floating point number *N* such that *a ≤ N ≤ b*

- `math` module

  `sqrt(num)`: Return the square root of *num*