Karel

Base Karel Commands

```
move()
turn_left()
put_beeper()
pick_beeper()
```

If Statements

```
if condition:
    code to execute if condition is true

if condition:
    code to execute if condition is true
else:
    code to execute if condition is false
```

Function Declaration

```
def my_function():
    # Function contents here
```

Karel Program Structure

```
# other helper functions you define

def main():
    # code to execute
```

Conditions

```
front_is_clear()        front_is_blocked()
beepers_present()       no_beepers_present()
beepers_in_bag()        no_beepers_in_bag()
left_is_clear()         left_is_blocked()
right_is_clear()        right_is_blocked()
facing_north()          not_facing_north()
facing_south()          not_facing_south()
facing_east()           not_facing_east()
facing_west()           not_facing_west()
```

Loops

```
for i in range(num_iterations):
    # code to execute in loop

while condition:
    # code to execute in loop
```

Common Python Functions

```
# prompts the user and returns the user-supplied text
user_input = input("prompt")

# prints the text to the console
print("Hello, World!")

# data-type conversion functions
# int(val), float(val), str(val)
```

Common List Functions

```
# adds a value to the end of a list
lst.append(value)

# removes and returns value at index, or at end if no index given
lst.pop(index)

# checks if a value is in a list
if val in lst:
     # body of if statement

# access a value at a given index
val = lst[index]
```

Common String Functions

```
# concatenates two strings together, and returns a new string
s = “CS” + “106A”

# accesses the character at a given index
char = s[index]

# checks if a substring is within a string
if “Hello” in s:
    # body of if statement

# tokenizes the string into its space-separated components
tokens = s.split()
```