Python to C

Updated: 14th August, 2019

This document it to demonstrate how to convert between python3 and C syntax. While they have many common attributes, they are still vastly different.

1   Basic Functionality

1.1   Hello World

Python

```python
def func()
    print("Hello World!")
```

C

```c
void func()
{
    printf("Hello World!\n");
}
```

1.2   User Input

Python

```python
def userInput():
    # Integer input
    var_int = int(input())

    # Double (real number)
    var_dbl = float(input())

    # String
    var_str = input()
```

C

```c
void userInput()
{
    int var_int;
    double var_dbl;
    char var_str[101];
    /* String with max length of
        100 characters */

    /* Integer */
    scanf("%d", &var_int);

    /* Double */
    scanf("%f", &var_dbl);

    /* String */
    scanf("%s", var_str);
    /* Note that no & is needed
        since its an array*/
}
```

2 Control Structures

2.1 If statement

Python

```python
def func(a):
   if a == 10:
      print("It is 10")
   elif a > 10:
      print("Greater than 10")
   else:
      print("Not high enough")
```

C

```c
void func(int a)
{
   if(a == 10)
   {
      printf("It is 10");
   }
   else if(a > 10)
   {
      printf("Greater than 10");
   }
   else
   {
      printf("Not high enough");
   }
}
```

2.2 For Loop

Python

```python
def func():
   # 0-10 Inclusive
   for ii in range(11):
      print(ii)
```

C

```c
void func()
{
   /* 0-10 Inclusive */
   int ii;
   for(ii = 0; ii < 11; ii++)
   {
      printf("%d\n", ii);
   }
}
```

2.3 While Loop

Python

```python
def func():
    var = 0

    while var >= 0:
        print("Enter value")
        var = int(input())
```

C

```c
void func()
{
    int var = 0;
    while( var >= 0)
    {
        printf("Enter value\n");
        scanf("%d", &var);
    }
}
```

2.4 Switch Statement

Python

```python
# Note that python has no actual
switch statement
def func(grade):
    if grade == 'A':
        print("Excellent")
    elif grade == 'B' or
          grade == 'C':
        print("Well Done")
    elif grade == 'D':
        print("You Passed")
    elif grade == 'F':
        print("Try again")
    else:
        print("Invalid Grade")
```

C

```c
void func(char grade)
{
    switch(grade)
    {
        case 'A':
            printf("Excellent");
            break;
        case 'B' : case 'C':
            printf("Well Done");
            break;
        case 'D':
            printf("You Passed");
            break;
        case 'F':
            printf("Try again");
            break;
        default:
            print("Invalid Grade")
    }
}
```

3   Function Calls

3.1   Main

Python	C

```python
if __name__ == "__main__":
    print("Argument Count:",
            len(sys.argv))

    print("Arguments: ")
    for arg in sys.argv:
        print(arg)
```

```c
int main(int argc, char *argv[])
{
    int ii;
    printf("Argument Count: %d\n",
            argc);

    printf("Arguments: ");
    for(ii = 0; ii < argc; ii++)
    {
        printf("%s\n", argv[ii]);
    }

    return 0;
}
```

3.2   Returning Values

Python	C

```python
def func3(x):
    # Do something

def func2():
    return 5

def func1():
    a = func2()
    func3(a)
```

```c
void func3(int x)
{
    /* Do something */
}

int func2()
{
    return 5;
}

void func1()
{
    int a = func2();
    func3(a);
}
```

Note: Data Structures such as Lists and Dicts do not naturally exist in C, therefore do not try to use them. Linked Lists will be covered later in this unit though.

End of Worksheet