Hello (1A)

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice.

While loop

Declaring a variable is simple enough: You enter the variable's type, some whitespace, the variable's name, and a semicolon:

```
double x;
```

In C, variable declarations belong at the top of the function in which they are used.

http://www.cs.toronto.edu/~patitsas/cs190/c_for_python.html

Variable declaration

To a Python programmer, it seems a pain to have to include these variable declarations in a program, though this gets easier with more practice. C programmers tend to feel variable declarations are worth the minor pain. The biggest advantage is that the compiler will automatically identify any time a variable name is misspelled, and point directly to the line where it is misspelled. This is a lot more convenient than executing a program and finding that it has gone wrong somewhere because of the misspelled variable name.

http://www.cs.toronto.edu/~patitsas/cs190/c_for_python.html

White spaces

In Python, whitespace characters like tabs and newlines are important: You separate your statements by placing them on separate lines, and you indicate the extent of a block (like the body of a while or if statement) using indentation. These uses of whitespace are idiosyncrasies of Python. (Admittedly, FORTRAN and BASIC also use line breaks to separate statements, but no other major language relies on whitespace for indicating blocks.)

http://www.cs.toronto.edu/~patitsas/cs190/c_for_python.html

White spaces

Python equivalent

```
disc = b * b - 4 * a * c
if disc < 0:
    num_sol = 0
else:
    t0 = -b / a
    if disc == 0:
        num_sol = 1
        sol0 = t0 / 2
    else:
        num_sol = 2
        t1 = disc ** 0.5 / a
        sol0 = (t0 + t1) / 2
        sol1 = (t0 - t1) / 2
```

C fragment
Python equivalent

```
disc = b * b - 4 * a * c;
if (disc < 0)
{
    num_sol = 0;
}
else
{
    t0 = -b / a;
    if (disc == 0)
    {
        num_sol = 1;
        sol0 = t0 / 2;
    }
    else
    {
        num_sol = 2;
        t1 = sqrt(disc) / a;
        sol0 = (t0 + t1) / 2;
        sol1 = (t0 - t1) / 2;
    }
}
```

http://www.cs.toronto.edu/~patitsas/cs190/c_for_python.html

print

print("v=",3,"cm :",x,",",y+4)
print options:
    sep=" " items separator, default space
    end="\n" end of print, default new line
    file=sys.stdout print to file, default standard output

s = input("Instructions:")
    input always returns a string, convert it to required type
(cf. boxed Conversions on the other side).

https://perso.limsi.fr/pointal/_media/python:cours:mementopython3-english.pdf

functions

Python equivalent

```
def gcd(a, b):
  if b == 0:
    return a
  else:
    return gcd(b, a % b)

print("GCD: " + str(gcd(24, 40)))
```

C fragment
C program

```
int gcd(int a, int b)
{
  if (b == 0)
  {
    return a;
  }
  else
  {
    return gcd(b, a % b);
  }
}

int main()
{
  printf("GCD: %d\n",
    gcd(24, 40));
  return 0;
}
```

http://www.cs.toronto.edu/~patitsas/cs190/c_for_python.html

References

[1] Essential C, Nick Parlante
[2] Efficient C Programming, Mark A. Weiss
[3] C A Reference Manual, Samuel P. Harbison & Guy L. Steele Jr.
[4] C Language Express, I. K. Chun