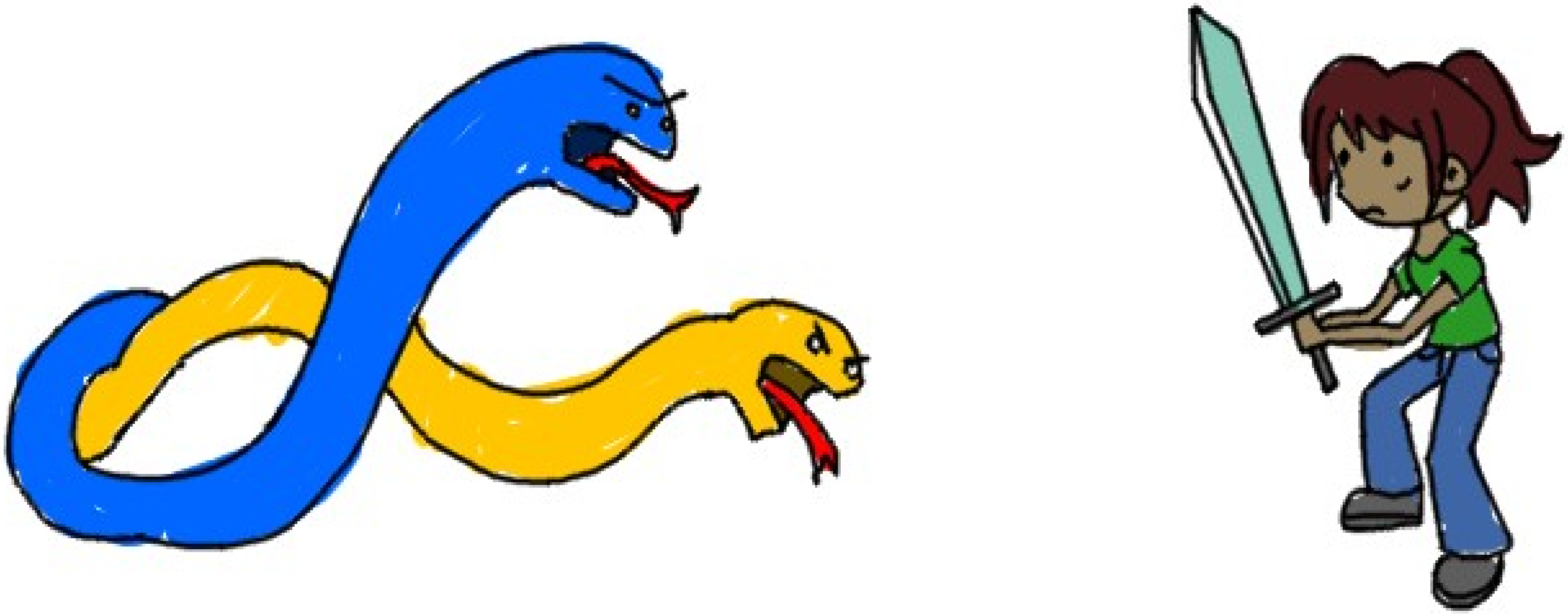
Python vs. C++



Written by Rachel J. Morris, last updated 2013-05-24

Undefined Behavior and Garbage

- In Effective C++ by Scott Meyers, he talks about **undefined behavior**:
  - *Programmers coming to C++ from languages like Java or C# may be surprised at the notion of* ***undefined behavior****. For a variety of reasons, the behavior of some constructs in C++ is literally not defined: you can't reliably predict what will happen at runtime.*
- You can write code in C++ that is unpredictable! This isn't a feature; it's something you need to be aware of.

Undefined Behavior and Garbage

- When you create a new variable in C++, it is not guaranteed to be initialized to a “safe” value.
- For example, if you had a variable pointing to a memory address, it would be initialized as **garbage**.
- If you try to access data outside of an array's predetermined size, you may get **garbage**.
- Garbage is generally a place in memory that is not in use (ie, outside of an array) and therefore may contain data being used by another program. To the user, it will look like *random garbage.*

Garbage!

```
#include <iostream>
using namespace std;

int main()
{
    int myArray[3] = { 2, 4, 6 };

    for ( int i = 0; i < 5; i++ )
    {
        cout << i << ": " << myArray[i] << ", \t";
    }
    cout << endl;

    return 0;
}
```

This program creates an array of size 3.

We iterate through the array, but we go from 0 to 5 (0, 1, 2, 3, 4).

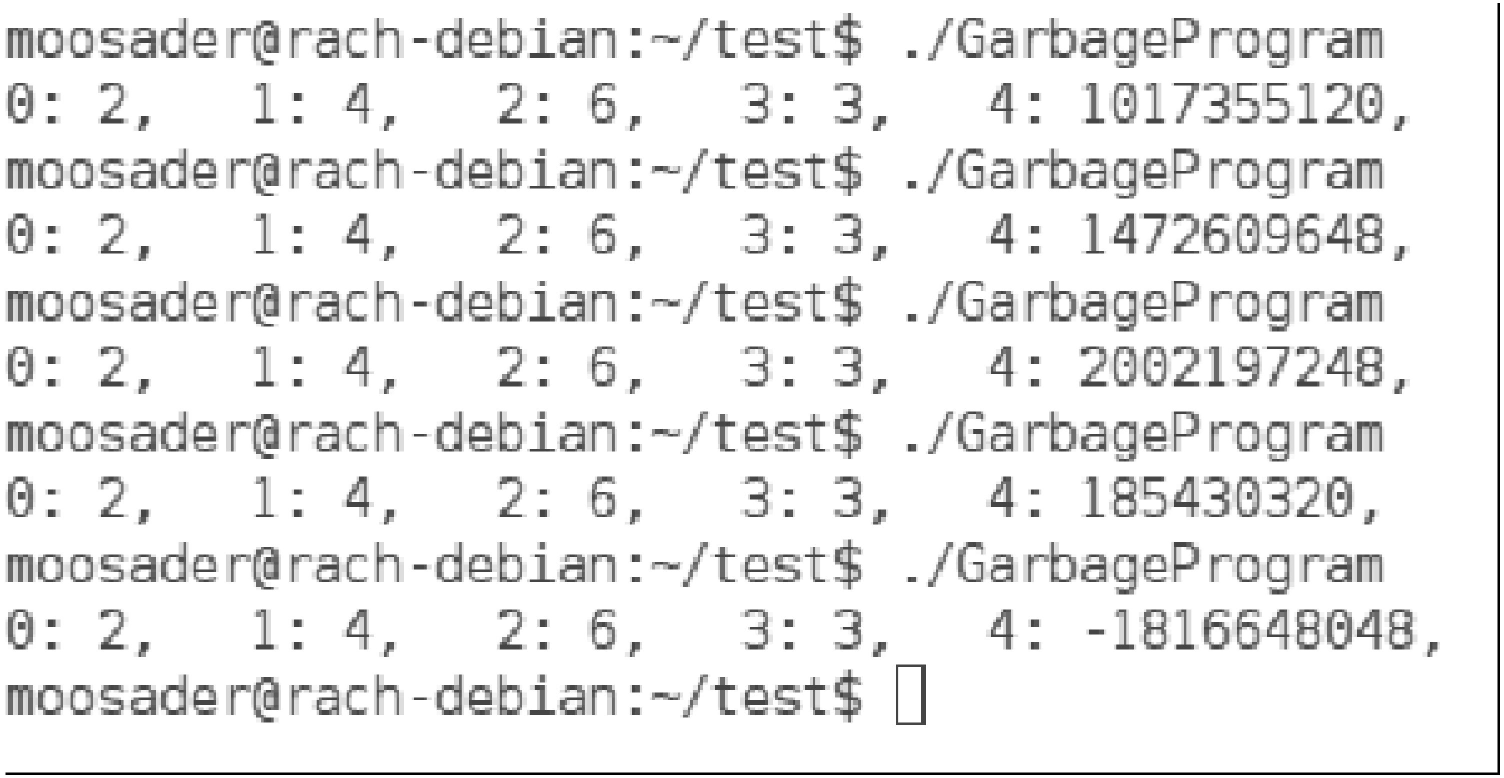
myArray[3] and myArray[4] are indices outside of the array.

Garbage!

Remember our array's values:

```
int myArray[3] = { 2, 4, 6 };
```

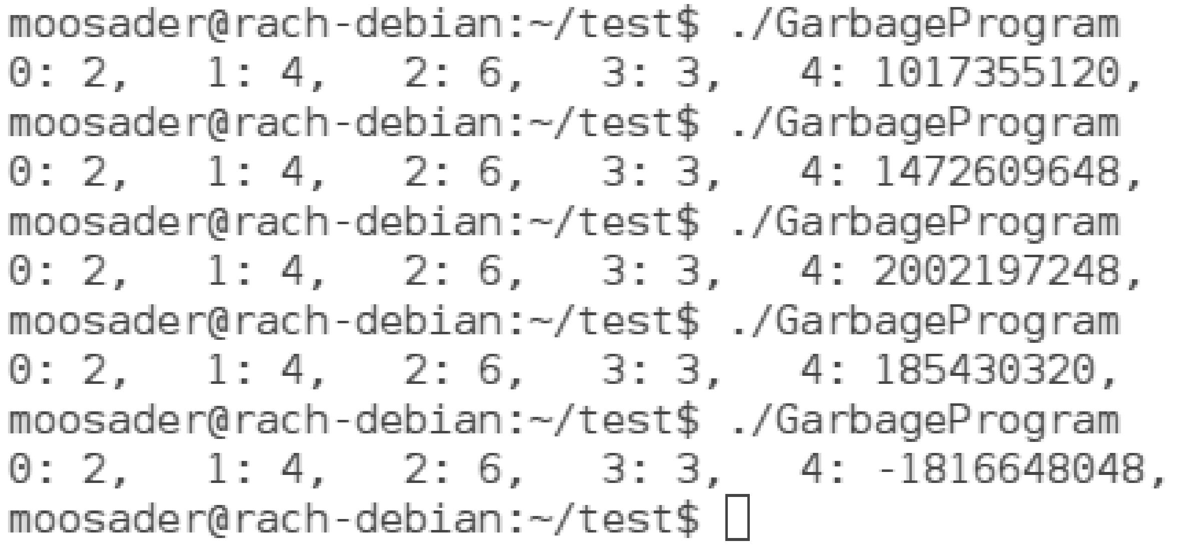
The program is run multiple times. 3 and 4 are garbage, even though 3's value is prettier. Notice that 4's value changes each time.



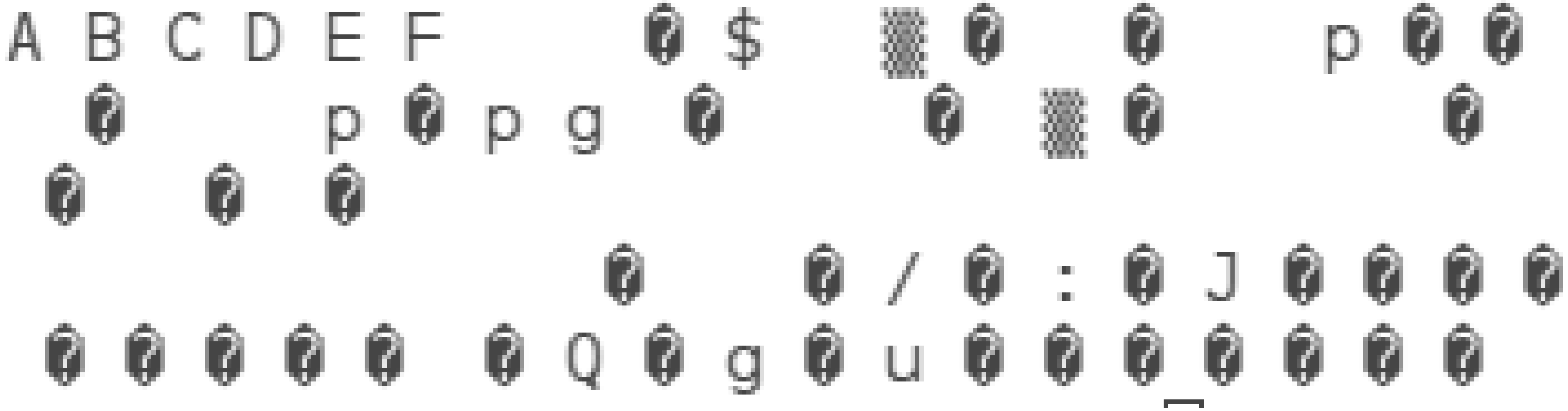
```
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 1017355120,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 1472609648,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 2002197248,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 185430320,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: -1816648048,
moosader@rach-debian:~/test$ 
```

Written by Rachel J. Morris, last updated 2013-05-24

Garbage!



```
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 1017355120,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 1472609648,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 2002197248,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: 185430320,
moosader@rach-debian:~/test$ ./GarbageProgram
0: 2,   1: 4,   2: 6,   3: 3,   4: -1816648048,
moosader@rach-debian:~/test$ ▯
```



```
A B C D E F     � $   ▒ �   �     p � �
 �     p � p g �     � ▒ �       �
�   �  �
                 �  � / � : � J � � � �
� � � � �  � Q � g � u � � �_� � � �
```

Written by Rachel J. Morris, last updated 2013-05-24

Avoid Garbage & Undefined Behavior

- Check whether an index is OK before accessing that index of an array.
- Initialize variables on declaration.
- Responsible memory management

Written by Rachel J. Morris, last updated 2013-05-24