NOTE: SCENARIO IS WHAT THE PUZZLE TAKER SEES.

SCENARIO

Consider a Python program, MovieHandler, that parses XML documents about movies. A snippet of its code is shown below. A user wants to parse `movie.xml` using this program. The program relies on the method `xml.sax.parse(filename_or_stream, handler[error_handler])`, which creates a SAX (Simple API for XML) parser to process the document. Consider the file movie.xml and the snippet of code below representing MovieHandler and answer the following questions, assuming that the code has all the required permissions to execute.

```
<!-- movie.xml -->
<!DOCTYPE root [
<!ENTITY do "do !">
<!ENTITY do2 "&do; &do;">
<!ENTITY do3 "&do2; &do2;">
<!ENTITY do4 "&do3; &do3;">
<!ENTITY do5 "&do4; &do4;">
<!ENTITY do6 "&do5; &do5;">
<!ENTITY do7 "&do6; &do6;">
<!ENTITY do8 "&do7; &do7;">
<!ENTITY do9 "&do8; &do8;">
<!ENTITY do10 "&do9; &do9;">
<!ENTITY do11 "&do10; &do10;">
<!ENTITY do12 "&do11; &do11;">
<!ENTITY do13 "&do12; &do12;">
<!ENTITY do14 "&do13; &do13;">
<!ENTITY do15 "&do14; &do14;">
<!ENTITY do16 "&do15; &do15;">
]>
<root>&do16;</root>
```

```
# import whatever is needed

class MovieHandler(xml.sax.ContentHandler):
    def startElement(self, name, attrs):
        for(attributes, value) in attrs.items():
            print attributes + " " + value

parser = xml.sax.make_parser()
parser.setContentHandler(MovieHandler())
parser.parse(open("movie.xml", "r"))
```

Questions:
1. What will the program do when executed?

2. If the running time to parse `movie.xml` and instantiate the objects is 10 ms, what would be the running time to parse the `movie-double.xml` file that has the same pattern of data but more entities, going up to `do32` and the root element is set to `<root>&do32;</root>`? (Choose the best answer)
a. The running time will be exponential
b. The running time would be about the same.
c. The running time would be less than 100 ms.
d. The running time would be slightly longer than 100 ms.
e. The running time would be slightly longer than 1000 ms.

[Other statistical questions will be imported here while creating the survey.]

NOTE: ANSWER IS TO BE SHOWN TO THE PUZZLE TAKER AT THE END OF THE SESSION.

ANSWERS:
1. The program will print out each of the elements of the XML document.

2. a
Because the subsequent entities are each composed of 2 of the previously defined entities, the running time will be exponential in the number of elements.

NOTE: THE REST OF THIS DOCUMENT CONTAINS EXTRA INFORMATION FOR THE PROJECT RESEARCHERS. IT IS NOT TO BE SHOWN TO THE PUZZLE TAKERS.

TAGS:
python, xml, billion-laughs, parsing

CATEGORIES:
Blindspot - YES
Type - Injection
Number of distinct functions - 7
Number of total functions - 7
Blindspot function - `parse()`
Function call omitted - NO
Blindspot type - Validation misuse
Number of parameters in the blindspot function - 1 parameter
Cyclomatic complexity - 2

NAME:
xml.sax.make_parser([parser_list])

DESCRIPTION:
Here are the details of the parameters −

- xmlfile: This is the name of the XML file to read from.
- contenthandler: This must be a ContentHandler object.
- errorhandler: If specified, errorhandler must be a SAX ErrorHandler object.

BLINDSPOT:
This XML processing module is not secure against maliciously constructed data. An attacker can abuse this type of vulnerability to perform denial of service attacks, access local files, make network connections to other machines, or circumvent firewalls. The attacks on XML abuse unfamiliar features, such as inline DTD (document type definition) with entities.

CORRECT USE EXAMPLE:
#N/A

MORE INFORMATION:
#N/A

REFERENCES:
1. https://docs.python.org/2/library/xml.html
2. https://en.wikipedia.org/wiki/Billion_laughs