

Concepts of Programming Languages, CSCI 305, Fall 2021
Categories of errors class exercise, Aug. 30

The following code snippets contain errors. For each of the code snippets tell if the error is a:

- a) syntax error detected by the lexical analyzer
- b) syntax error detected by the parser
- c) static semantic analysis error
- d) dynamic semantic analysis error
- e) compiler can't catch, may be caught by the hardware

1.

```
void main ( ) {  
    int x=0;  
    x=x+y;  
}
```

y is not declared - static semantic analysis error, c.

2.

```
void main ( ) {  
    in a, b, c;  
}
```

Syntax error detected by the parser, b.

3.

```
void main ( ) {  
    int b[ ] = { 1, 2, 3 };  
    int index = 9;  
    int a;  
    a = b[index];  
}
```

Array index is out of bounds. Checking out of bound indices can be turned on or off. In C it is usually turned off, so this would not be detected and may or may not cause an error. In Java and C# this is on so would be a dynamic semantic analysis error (an error caught at run time, but by check inserted into the code by the compiler). (In this simple case the compiler might have been able to catch the error, but in general, catching an error like this requires code added to the program.) Good answers – d and e.