1. A stack is a data structure in which items are inserted at one end and must be removed from the same end. As an analogy, think of a stack of trays in a cafeteria. When you add new trays you push the whole stack down and when you remove a tray the stack pops up. Implement a Stack object with the following methods:

- `__init__` which creates an empty stack
- `__str__` which returns a string representing the contents of the stack
- `empty` which tests whether the stack is empty
- `push` which adds an item to the top of the stack
- `pop` which removes and returns the item at the top of the stack
- `size` which returns the number of items on the stack

You can test your class definition with the following program:

```python
def main():
    s = Stack()
    if s.empty():
        print "Stack is empty"
    s.push('a')
    print s
    s.push('b')
    print s
    s.push('c')
    print s
    print s.size()
    print s.pop()
    print s
```

This program should produce the output:

```
Stack is empty
Stack: a
Stack: b a
Stack: c b a
3
c
Stack: b a
```

2. Explain the following object-oriented concepts and cite specific examples from the previous problem to illustrate your points.

(a) method
(b) instance variable
(c) instance of a class
(d) constructor
(e) accessor
(f) mutator