05 - Intro Recursion (Real Life)

Two parts:

1) Terminating condition/base case
2) Recursive case

Real Life Examples:

- **Family Tree**

```
To draw the family tree of a person
     write the name of the person
     if the person had children
          for each child
               draw the family tree of the child
```

- **Climbing Stairs**

```
To climb the stairs
     if you are at the top
          you are done
     else
          take one step up
          climb the stairs
```

- **Reading a book** (if at end - stop, if not - next page), **Files**

Math Examples:

- **Sequence**

$$t_1 = 3$$
$$t_n = 2 \times t_{n-1}, \quad if\ n > 1$$

- **Euclid**

1. If m = n, then gcd(m, n) = m
2. If m > n, then gcd(m, n) = gcd(n, m - n)
3. If m < n, then gcd(m, n) = gcd(n, m)

```
public static int gcd (int m, int n)
{
     if (m == n)
          return m;
     else if (m > n)
          return gcd(n, m - n);
     else
          return gcd(m, n - m);
}
```

06 - Simple Recursive Algorithms

Recursion with Strings

Reverse String Method

```
public static String reverseString (String s)
{
     if (s.length() > 0)
          return reverseString(s.substring(1)) + s.charAt(0);
     else
          return "";
}
```

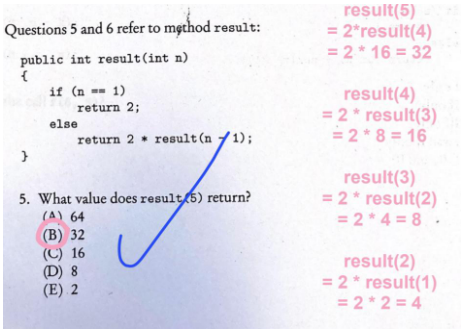
Permutations Method

```
public static void permuteString (String s, int index)
{
    String nextString;
    if (index == s.length())
    {
        System.out.println(s);
    }
    else
    {
        for (int i = index; i < s.length(); i++)
        {
            nextString = s.substring(0,index)
                         + s.charAt(i)
                         + s.substring(index,i)
                         + s.substring(i+1);

            permuteString(nextString, index+1);
        }
    }
}
```

AP 7 - Recursion

Trace Example



Questions 5 and 6 refer to method result:

```
public int result(int n)
{
    if (n == 1)
        return 2;
    else
        return 2 * result(n - 1);
}
```

5. What value does result(5) return?
   (A) 64
   (B) 32
   (C) 16
   (D) 8
   (E) 2

result(5)
= 2*result(4)
= 2 * 16 = 32

result(4)
= 2 * result(3)
= 2 * 8 = 16

result(3)
= 2 * result(2)
= 2 * 4 = 8

result(2)
= 2 * result(1)
= 2 * 2 = 4