Tutorial 6 – Recursive Functions – Suggested Answers

Q1: (rSumup)

```
int rSumup1(int n)
{
  if (n == 1)
    return 1;
  else
    return n +  rSumup1(n-1);
}
void rSumup2(int n, int *result)
{
  if (n == 1)
    *result=1;
  else
  {
    rSumup2(n-1, result);
    *result += n;
  }
}
```

Q2: (rDigitValue)

```
int rDigitValue1(int num, int k)
{
  if (k==0)
    return 0;
  else if (k==1)
    return num%10;
  else
    return rDigitValue1(num/10, k -1);
}
void rDigitValue2(int num, int k, int *result)
{
  if (k==0)
    *result = 0;
  else if (k==1)
    *result = num%10;
  else
    rDigitValue2(num/10, k-1, result);
}
```

Q3:

If we execute the program and input to the program “ward ” then the output will be “draw”:

Program Output
Enter your word and end it with a space => ward draw $

Please note that there is a blank character at the end of the input word before the “enter” key is pressed.
Basically, this program prints an input string, which ends with a space character, in the reversed order. (tutors: please explain why)

Q4: (rCountArray)

```
int rCountArray(int array[], int n, int a)
{
  if  (n == 1)
  {
    if (array[0] == a)
      return 1;
    else
      return 0;
  }
  if (array[0] == a)
    return 1 + rCountArray(&array[1], n-1, a);
  else
    return rCountArray(&array[1], n-1, a);
}
/* another version
int rCountArray(int array[], int n, int a)
{
  if  (n == 1)
  {
    if (array[0] == a)
      return 1;
    else
      return 0;
  }
  if (array[n-1] == a)
    return 1 + rCountArray(&array[0], n-1, a);
  else
    return rCountArray(&array[0], n-1, a);
} */
/* another version
int rCountArray(int array[], int n, int a)
{
  int count;

  if(n == 0)
    return 0;
  count = rCountArray(array + 1, n - 1, a);
  if(*array == a)
    return count + 1;
  else
    return count;
}
*/
```