QUESTION 1

```
/* Question 1 Solution */
#include <stdio.h>

int main( void )
{
   int salaries[ 11 ] = { 0 }; /* array to hold salary counts */
   int sales; /* current employee's sales */
   double salary; /* current employee's salary */
   double i = 0.09; /* commission percentage */

   /* prompt user for gross sales */
   printf( "Enter employee gross sales ( -1 to end ): " );
   scanf( "%d", &sales );

   /* while sentinel value not read from user */
   while ( sales != -1 ) {

      /* calculate salary based on sales */
      salary = 200.0 + sales * i;
      printf( "Employee Salary is $%.2f\n", salary );

      /* update appropriate salary range */
      if ( salary >= 200 && salary < 1000 ) {
         ++salaries[ ( int ) salary / 100 ];
      } /* end if */
      else if ( salary >= 1000 ) {
         ++salaries[ 10 ];
      } /* end else if */

      /* prompt user for another employee sales amount */
      printf( "\nEnter employee gross sales ( -1 to end ): " );
      scanf( "%d", &sales );
   } /* end while */

   /* display table of ranges and employees in each range */
   printf( "\nEmployees in the range:\n" );
   printf( "$200-$299 : %d\n", salaries[ 2 ] );
   printf( "$300-$399 : %d\n", salaries[ 3 ] );
   printf( "$400-$499 : %d\n", salaries[ 4 ] );
   printf( "$500-$599 : %d\n", salaries[ 5 ] );
   printf( "$600-$699 : %d\n", salaries[ 6 ] );
   printf( "$700-$799 : %d\n", salaries[ 7 ] );
   printf( "$800-$899 : %d\n", salaries[ 8 ] );
   printf( "$900-$999 : %d\n", salaries[ 9 ] );
   printf( "Over $1000: %d\n", salaries[ 10 ] );

   return 0; /* indicate successful termination */
} /* end main */
```

QUESTION 2

```
/* Question 2 Solution */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main( void )
{
   long i; /* loop counter */
   int j; /* loop counter */
   int x; /* first die */
   int y; /* second die */
   int sum[ 13 ] = { 0 }; /* count occurrences of each combination */

   /* array expected contains counts for the expected number of times each sum occurs in
36 rolls of the dice */
   int expected[ 13 ] = { 0, 0, 1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1};


   srand( time( NULL ) ); /* seed random number generator */

   /* roll dice 36,000 times */
   for ( i = 1; i <= 36000; i++ ) {
      x = 1 + rand() % 6;
      y = 1 + rand() % 6;
      ++sum[ x + y ];
   } /* end for */

   printf( "%10s%10s%10s%10s\n", "Sum", "Total", "Expected", "Actual" );

   /* display results of rolling dice */
   for ( j = 2; j <= 12; j++ ) {
      printf( "%10d%10d%9.3f%%%9.3f%%\n", j, sum[ j ],
         100.0 * expected[ j ] / 36, 100.0 * sum[ j ] / 36000 );
   } /* end for */

   return 0; /* indicate successful termination */
} /* end main */
```