```c
/*
 *
 * COEN 11
 * Homework 4 Unions
 * Answers
 *
 */

#include <stdio.h>

#define M   20
#define SIZE 3

/*
 * Types
 */

union union_name
{
    int y;
    char s[M];
};

struct struct_name
{
    int x;  //0 - string s is valid, 1 - integer y is valid
    union union_name u;
};

/*
 * Global variables
 */

struct struct_name array[M];

/*
 * Question 1
 * Count number of elements y between 10 and 100,
 * and number of elements y that are either less than 10 or more
than 100.
 */

void count (void)
{
    int i, counter = 0, counter2 = 0;
```

```
    for (i = 0; i < M; i++)
    {
     if (array[i].x == 1)     //y is valid
     {
         if (array[i].u.y > 10 && array[i].u.y < 100)
          counter++;
         else
          if (array[i].u.y < 10 || array[i].u.y > 100)
              counter2++;
     }
     return;
    }
}

/*
 * Question 2
 * Count number of strings with size greater than SIZE
 * Using pointers
 */

int count (void)
{
    int i, counter = 0, len = 0;
    char *p;
    struct struct_name *ptr;

    ptr = array;

    for (i = 0; i < M; i++, ptr++)
    {
     if (ptr->x == 0)    //string is valid
     {
         p = ptr->u.string;
         len = 0;
         while (*p != '\0')   // same as: len = strlen(p);
         {
          len++;
          p++;
         }
         if (len > SIZE)
          counter++;
     }
    }
    return counter;
}
```

```c
/*
 * Question 3
 * Output all the elements in the array
 * Version using index
 */

void output (void)
{
    int i;

    for (i = 0; i < M; i++)
    {
     printf ("%d\n", array[i].x);
     if (array[i].x == 0)
         printf ("%s\n", array[i].u.string);
     else
         printf ("%d\n", array[i].u.y);
    }
    return;
}

/*
 *  Question 3
 *  Version using pointers
 */

void output (void)
{
    int i;
    struct struct_name *ptr = array;

    for (i = 0; i < M; i++, ptr++)
    {
     printf ("%d\n", ptr->x);
     if (ptr->x == 0)
         printf ("%s\n", ptr->u.string);
     else
         printf ("%d\n", ptr->u.y);
    }
    return;
}
```