C Programs for Practice (Strings)

1. **C program to add or concatenate two strings**

```c
#include <stdio.h>
#include <string.h>

void concat(char *s1, const char *s2) {

    // Copy s2 to the end of s1 using memcpy()
    memcpy(s1 + strlen(s1), s2, strlen(s2) + 1);
}

int main() {
    char s1[50] = "Hello";
    char s2[] = "World";

    // Concatenate using memcpy()
    concat(s1, s2);

    printf("%s", s1);
    return 0;
}
```

2. **C Program to find the length of a string**

```c
#include <stdio.h>
#include <string.h>

int main() {
    char s[] = "Programming";

    // Find length of string using strlen()
    printf("%lu", strlen(s));

    return 0;
}
```

3. **C Program to check for palindrome string**

```c
// Program to check for a palindrome string by reversing
// the string
#include <stdio.h>
#include <string.h>
#include <stdbool.h>
```

```c
#include <stdlib.h>

char *strrev(char *str) {
   int len = strlen(str);

     // Temporary char array to store the
   // reversed string
   char *rev = (char *)malloc
     (sizeof(char) * (len + 1));

   // Reversing the string
   for (int i = 0; i < len; i++) {
      rev[i] = str[len - i - 1];
   }
   rev[len] = '\0';
   return rev;
}

void isPalindrome(char *str) {

   // Reversing the string
   char *rev = strrev(str);

   // Check if the original and reversed
     // strings are equal
   if (strcmp(str, rev) == 0)
      printf("\"%s\" is palindrome.\n",
            str);
   else
      printf("\"%s\" is not palindrome.\n",
            str);
}

int main() {

   // Cheking for palindrome strings
   isPalindrome("madam");
    isPalindrome("hello");

   return 0;
}
```

4. C Program to compare two strings

```c
#include <stdio.h>
#include <string.h>

int comp(char *s1, char *s2) {

    // Compare characters one by one
    while (*s1 && !(*s1 - *s2)) {
        s1++;
        s2++;
    }

    // Returning 0 if strings are equal
    // +ve value if s1 is greater than s2
    // -ve value if s1 is smaller than s2
    return *s1 - *s2;
}

int main() {
    char s1[] = "Hello";
    char s2[] = "Hello";

    // Comparing s1 and s2
    if(!comp(s1, s2))
        printf("Equal");
    else
        printf("Not Equal");

    return 0;
}
```

5. C Program to insert a string to another string

```c
#include <stdio.h>

void concat(char *s1, char *s2) {
    int i = 0;

    // Move to the end of str1
    while (s1[i] != '\0')
        i++;

    // Copy characters from str2 to str1
    int j = 0;
    while (s2[j] != '\0') {
```

```c
        s1[i] = s2[j];
        i++;
        j++;
    }

    // Null-terminate the concatenated string
    s1[i] = '\0';
}

int main() {
    char s1[50] = "Hello ";
    char s2[] = "World";

    concat(s1, s2);

    printf("%s", s1);
    return 0;
}
```