Assignment 64: What are the sprint () and sscanf() Function?

The sprintf() and sscanf() functions in C are used for formatted input/output operations, similar to printf() and scanf(), but they work with strings instead of the standard input and output streams.

❖ sprintf():

- sprintf() is used to format data and store it as a string.
- It takes a format string similar to printf() and writes the formatted data into a character array (string) instead of printing it to the standard output.
- The function signature is: int sprintf(char *str, const char *format, ...).

Example -

```c
char buffer[100];
int num = 10;
sprintf(buffer, "The value of num is %d", num);
printf("%s\n", buffer); // Output: The value of num is 10
```

❖ sscanf():

- sscanf() is used to extract data from a string based on a specified format.
- It takes a format string similar to scanf() and reads data from a character array (string) instead of the standard input.
- The function signature is: int sscanf(const char *str, const char *format, ...).

Example -

```c
char str[] = "The value of num is 10";
int num;
sscanf(str, "The value of num is %d", &num);
printf("num = %d\n", num); // Output: num = 10
```

Both sprintf() and sscanf() are useful for formatting data into strings and parsing data from strings, respectively. They provide a convenient way to work with formatted data without using the standard input/output streams. However, it's important to use them carefully to avoid buffer overflow and format mismatch issues.