11.- ft_strlcpy.-

Function based on the definition given in the BSD man pages for "strlcpy(3)".
The library associated is <string.h>.

Synopsis:

size_t strlcpy(char * restrict dst, const char * restrict src, size_t dstsize);

we shall use:

size_t ft_strlcpy(char *dst, const char *src, size_t dstsize);

Purpose

- Copies a string from one location to another, ensuring null-termination and preventing buffer overflows.

Parameters

- `dst`: Pointer to the destination string buffer.
- `src`: Pointer to the source string to be copied.
- `dstsize`: Size of the destination buffer, including space for the null terminator.

Return Value

- Returns the total length of the source string (`src`), including the null terminator.

Description

- Copies up to `dstsize - 1` bytes from `src` to `dst`.
- Always null-terminates the destination string, even if truncation occurs.
- Returns the total length of `src`, allowing the caller to determine if truncation happened.

The BSD Library Functions Manual says: The strlcpy() and strlcat() functions copy and concatenate strings with the same input parameters and output result as snprintf(3). They are designed to be safer, more consistent, and less error prone replacements for the easily misused functions strncpy(3) and strncat(3).

Code

```
#include "libft.h"

size_t ft_strlcpy(char *dst, const char *src, size_t dstsize)
{
    unsigned int i;
    unsigned int c;

    i = 0;
    c = 0;
    while (src[c] != '\0')
        c++;
    if (dstsize != 0)
    {
        while (src[i] != '\0' && i < (dstsize - 1))
        {
            dst[i] = src[i];
```

```
            i++;
        }
        dst[i] = '\0';
    }
    return (c);
}
```

Code Explanation

1. **Counts source string length:**
   - Iterates through `src` to count its total length, including the null terminator (`c`).
2. **Copies characters (if space allows):**
   - If `dstsize` is not zero:
     - Copies characters from `src` to `dst` until:
       - The end of `src` is reached.
       - The available space in `dst` is filled (`dstsize - 1`).
   - Ensures null-termination of `dst`.
3. **Returns source string length:**
   - Returns the total length of `src` (`c`), regardless of truncation.

Main Function (Optional)

```
int main(void)
{
    char src[10] = "Bye World!";
    char dst[4];
    size_t len = ft_strlcpy(dst, src, sizeof(dst));

    printf("Copied string: %s\n", dst);
    printf("Original string length: %zu\n", len);
    return (0);
}
```

The main function shows how it works; giving as the expected return (src length) and copied string in dst.