16 .- ft_strrchr.-

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

Synopsis:

```
char *strrchr(const char *s, int c);
```

Purpose:

Locates the last occurrence of a character (c) within a string (s).

Parameters:

- s: The string to search within.
- c: The character to search for.

Return Value:

Returns a pointer to the last occurrence of c in s, or NULL if c is not found.

Description:

- Iterates through the string from beginning to end, keeping track of the last occurrence of c.
- Returns a pointer to the last matching character's position in s.

Code

```c
#include "libft.h"

char *ft_strrchr(const char *s, int c)
{
    char str = c;
    char *end = NULL;
    int i = 0;

    while (s[i] != '\0')
    {
        if (s[i] == str)
            end = (char *)&s[i];
        i++;
    }
    if (s[i] == (unsigned char)c)  // Check for null terminator as well
        return ((char *)&s[i]);
    return (end);
}
```

Code Explanation

1. **Converts c to char:** Stores c as a character for comparison (str).
2. **Initializes variables:** Sets end to NULL to track the last occurrence.
3. **Iterates through string:**
   - Checks each character of s against str.
   - Updates end if a match is found.
4. **Checks null terminator:**
   - Explicitly checks if the null terminator itself matches c.

5. **Returns pointer or NULL:**
   - Returns `end` (pointer to the last match) or `NULL` if no match was found.

Main Function (Optional)

```
int main(void)
{
    char *str = "Hello, friend";
    char c = 'i';
    char *ptr;
    int n;

    ptr = ft_strrchr(str, c);
    n = ft_strlen(ptr);
    if (ptr)
    {
        printf("Character '%c' found in string '%s' back position '%d'.\n",
            c, str, n);
    }
    else
    {
        printf("Character '%c' not found in string '%s'.\n", c, str);
    }
    return (0);
}
```

Key Points:

- **Tracking Last Occurrence:** The `end` pointer stores the position of the last match.
- **Checking Null Terminator:** The extra check for `s[i] == (unsigned char)c` ensures the null terminator is considered as a potential match.
- **Character Comparison:** Uses `==` to compare characters directly.