3.- `ft_isalnum.-`

Function based on the definition given in the BSD man pages for “isalnum”.
The library associated is <ctype.h>.

Sinopsis: int isalnum (int c);

Purpose: Checks for an alphanumeric character.

Parameters:

- `c`: The character to be checked.

Return value:

- 1 if `c` is an alphanumeric character, and 0 otherwise.

Description:

The **`ft_isalnum`** function checks whether the given character is an alphanumeric character. An alphanumeric character is any character from the set **`['A'-'Z', 'a'-'z', '0'-'9']`**. The function uses a logical `OR` operator to check if the character's ASCII value falls within one of the three ranges: uppercase letters (65-90), lowercase letters (97-122), or digits (48-57). If the character falls within any of these ranges, it returns 1; otherwise, it returns 0.

Code:

```
#include "libft.h"
int   ft_isalnum(int c)
{
      return ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z')
            || (c >= '0' && c <= '9'));
}
/*
int   main(void)
{
      char  c;

      printf("Input an alphanumeric character to obtain 1: ");
      scanf("%c", &c);
      printf("%d\n", ft_isalnum(c));
      return (0);
}
*/
```

Code explanation:

1. **Include header file:** The `#include "libft.h"` statement includes the header file **`libft.h`**, which defines the required libraries for our function.
2. **Define function:** The `int ft_isalnum(int c)` statement defines the `ft_isalnum` function. The function takes one argument, `c`, which is the character to be checked.
3. **Return value:** The **`return ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z') || (c >= '0' && c <= '9'));`** statement returns 1 if `c` is an alphanumeric character, and

0 otherwise. This statement checks if the ASCII value of `c` falls within any of the three ranges: uppercase letters (65-90), lowercase letters (97-122), or digits (48-57). If the character falls within any of these ranges, the statement returns 1; otherwise, it returns 0.

4. Under comments we develop a main function to show how it works:

4.1. Main function: The **`int main(void)`** statement defines the main function, which is the entry point of the program.

4.2. Input and check alphanumeric: The **`scanf("%c", &c);`** statement reads a character from the user's input and stores it in the variable **`c`.** The printf `(ft_isalnum(c))` statement checks whether **`c`** is an alphanumeric character. If **`c`** is an alphanumeric character, the statement prints 1 indicating that **`c`** is an alphanumeric character; otherwise, it prints 0 indicating that **`c`** is not an alphanumeric character.

4.3. Return value: The **`return (0);`** statement exits the program with a status code of 0, indicating that the program executed successfully.