1.- ft_isalpha.-

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

Sinopsis: int isalpha (int c);
Parameters: int c.

Description: checks for an alphabetic character; in the standard "C" locale, it is equivalent to (isupper(c) || islower(c)). The values returned are nonzero if the character c falls into the tested class, and zero if not.
An alphabetic character is any letter from the English alphabet, either uppercase (A-Z) or lowercase (a-z).

Code:

```
#include "libft.h"

int   ft_isalpha(int c)
{
      return ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z'));
}
/*
int   main(void)
{
      char  c;

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

Explanation:
1. We include the library **libft.h** where all the required libraries are included.
2. We declare the function according BSD man page sinopsis: **int ft_isalpha(int c).**
3. We declare the return function using a boolean (OR) where we shall obtain 1 or 0; 1 if any of the conditions is true it will give us 1; if no one of the condition is true then 0.
4. Under comments we develop a main function to show how it works:
    4.1 **int main(void)** - we declare an int function with void value.
    4.2 **char c;** – we declare the character c,
    4.3 **printf** – here we send a message for an input.
    4.4 **scanf**(“%c”, &c) – we request here the input using %c for a character and &c where our value will be stored.
    4.5 **printf** – will print the value %d (integer type) with the value of ft_isalpha(c).
4.6 **return (0)** – exits the program giving a zero value which means all went well.