Parser Combinators

Consider the following grammar,

```
  <expr> ::= a<B> | ϵ
     <B> ::= b<expr>
```

where ϵ stands for “nothing.”

A fellow programmer came up with the following parser, but it’s not working the way they want. They come to you for help.

```
let parse input =
    // parser
    let grammar = pmany0 (pstr "ab")

    // parse input
    match grammar (prepare input) with
    | Success(res,_) -> printfn "%A" res
    | Failure _ -> printfn "Invalid expression"
```

The parser correctly accepts everything the grammar says it should, like the strings `"ab"`, `"abab"`, `""`, and `"ababab"`. However, it also incorrectly accepts the following strings: `"aba"`, `"abfoobar"`, and `"Wait, how the heck is this possible?"`.

1. What is the cause of the bug in the above program?

2. There is a simple modification to the program of the form:

```
    let grammar = pleft (pmany0 (pstr "ab")) <p>
```

Use the supplied tables to find the parser `<p>` that fixes this bug.