41 .- Value [] for the final expression [] (Line 40) is not correct.

0 .- In the function:
correct_nums(Exp, Digits) ->
    case re:run(Exp, "([0-9]+)",
                [global, {capture, all_but_first, list}])
        of
      nomatch -> "No number entered";
      {match, IntLs} ->
          case [X
                || [X] <- IntLs,
                   not lists:member(list_to_integer(X), Digits)]
              of
            [] -> [];
            L -> L
          end
    end.
pattern of first clause succeed.
                              Is this correct

39 .- For the case expression:
case re:run(Exp, "([0-9]+)",
            [global, {capture, all_but_first, list}])
    of
  nomatch -> "No number entered";
  {match, IntLs} ->
      case [X
            || [X] <- IntLs,
               not lists:member(list_to_integer(X), Digits)]
          of
        [] -> [];
        L -> L
      end
end
Is there anything incorrect?
1.- The context:
        Digits = [9,2,2,3]
        Exp = "(9*2)+(2*3)"
2.- The argument value: {match,[["9"],["2"],["2"],["3"]]}.
3.- Enter in the second clause.
4.- The bindings:
        IntLs = [["9"],["2"],["2"],["3"]]
5.- The final value: [].
6.- Nothing.

40 .- For the case expression:
case re:run(Exp, "([0-9]+)",
            [global, {capture, all_but_first, list}])
    of
  nomatch -> "No number entered";
  {match, IntLs} ->
      case [X
            || [X] <- IntLs,
               not lists:member(list_to_integer(X), Digits)]
          of
        [] -> [];
        L -> L
      end
end
Is there anything incorrect?
1.- The context:
        Digits = [9,2,2,3]
        Exp = "(9*2)+(2*3)"
2.- The argument value: {match,[["9"],["2"],["2"],["3"]]}.
3.- Enter in the second clause.
4.- The bindings:
        IntLs = [["9"],["2"],["2"],["3"]]
5.- The final value: [].
6.- Nothing.

2 .- Given the context:
        Exp = "(9*2)+(2*3)",
The case expression:
case re:run(Exp, "([0-9]+)",
            [global, {capture, all_but_first, list}])
    of
  {match, IntLs} ->
      case [X
            || [X] <- IntLs,
               not lists:member(list_to_integer(X), Digits)]
          of
        [] -> [];
        L -> L
      end
end
matching with second clause succeed
case argument value :{match,[["9"],["2"],["2"],["3"]]}
Bindings:
                    IntLs = [["9"],["2"],["2"],["3"]]

1 .- Given the context:
        Exp = "(9*2)+(2*3)",
The case expression:
case re:run(Exp, "([0-9]+)",
            [global, {capture, all_but_first, list}])
    of
  nomatch -> "No number entered"
end
matching with first clause failed
case argument value :{match,[["9"],["2"],["2"],["3"]]}

37 .- For the case expression:
case [X
      || [X] <- IntLs,
         not lists:member(list_to_integer(X), Digits)]
    of
  [] -> [];
  L -> L
end
Is there anything incorrect?
1.- The context:
        Digits = [9,2,2,3]
        IntLs = [["9"],["2"],["2"],["3"]]
2.- The argument value: [].
3.- Enter in the first clause.
4.- The final value: [].
5.- Nothing.

38 .- For the case expression:
case [X
      || [X] <- IntLs,
         not lists:member(list_to_integer(X), Digits)]
    of
  [] -> [];
  L -> L
end
Is there anything incorrect?
1.- The context:
        Digits = [9,2,2,3]
        IntLs = [["9"],["2"],["2"],["3"]]
2.- The argument value: [].
3.- Enter in the first clause.
4.- The final value: [].
5.- Nothing.

36 .- Given the context:
        Digits = [9,2,2,3]
        IntLs = [["9"],["2"],["2"],["3"]],
The case expression:
case [X
      || [X] <- IntLs,
         not lists:member(list_to_integer(X), Digits)]
    of
  [] -> []
end
          matching with first clause succeed