1. Program prints `Enter expression:` and reads the whole input line into `expr`.
2. A `stringstream ss(expr)` is created so the program can extract tokens (numbers and operators) easily.
3. The first `int num` is read from the stream (`ss >> num`) and stored in `result` as the starting value.
4. The program enters a `while` loop that tries to read `op` (a `char`) and then another `num` from the stream.
5. Each loop iteration finds an operator (`+`, `-`, `*`, or `/`) followed by the next integer.
6. If `op` is `'+'` the program adds `num` to `result`.
7. If `op` is `'-'` it subtracts `num` from `result`.
8. If `op` is `'*'` it multiplies `result` by `num`.
9. If `op` is `'/'` it divides `result` by `num` using integer division (fractions are truncated).
10. The loop repeats until there are no more `op` + `num` pairs to read from the stream.
11. Finally, the program prints `Result =` followed by the final `result` value.
12. Whitespace in the input is ignored by the `stringstream`, and only the four operators above are handled (other chars are not processed).