Serialization protocol:
from client to server: general_stub is the “header” of the protocol: operator stores the indicator of different operations; stub_size stores the size of the content, which tells the server how much memory to allocate for the incoming protocol. operator_stub is the actual content of the protocol: a,b,c,offset and str are the parameter fields to be sent to the server.

```
typedef struct {
    int operator;
    size_t stub_size;
} general_stub;

typedef struct {
    int a;
    int b;
    int c;
    off_t offset;
    char str[];
} operator_stub;
```

from client to server: I use the same two struct for this one. It’s concise and easy to implement.

Note
I didn’t implement concurrency, so the server only processes one request at a time.