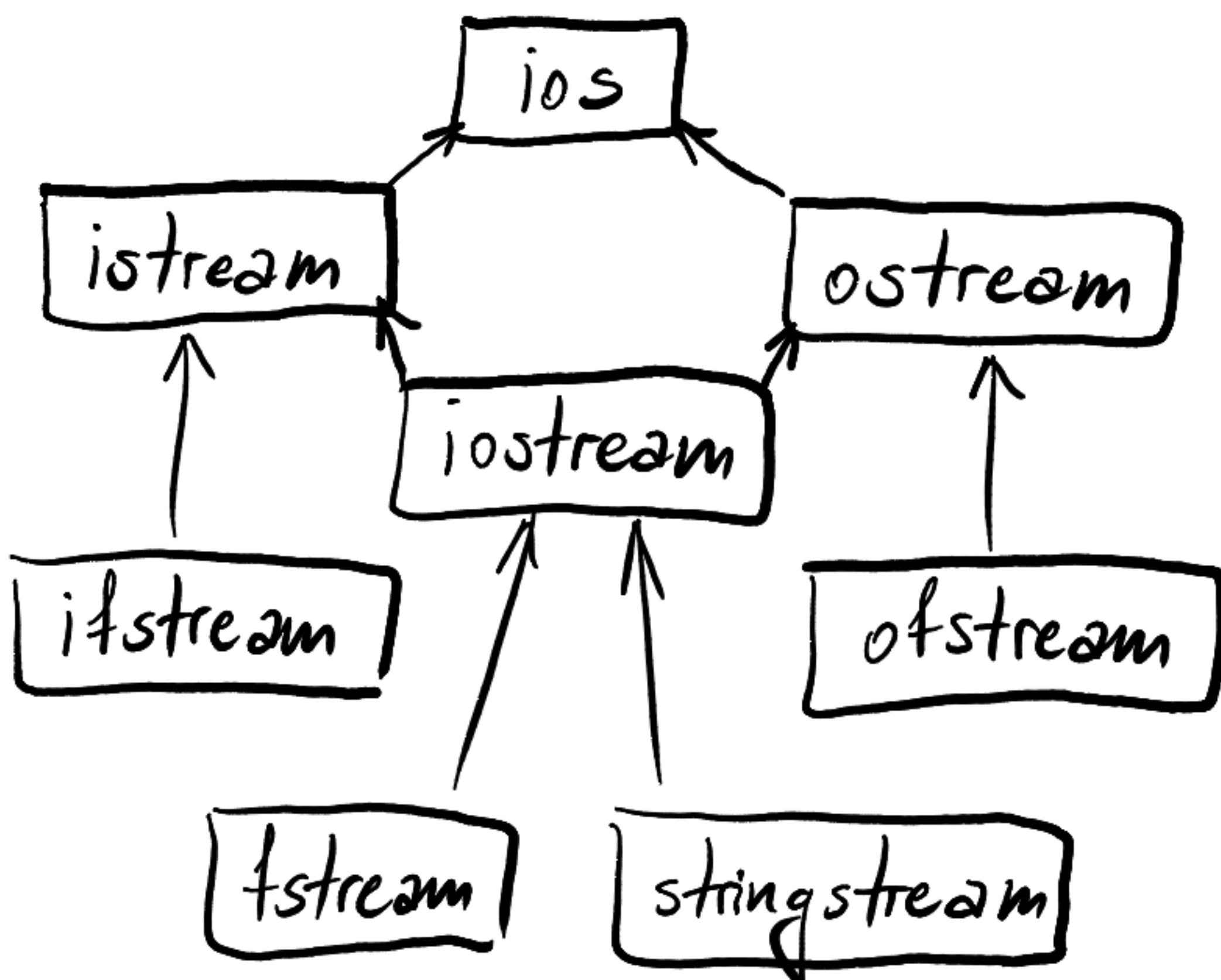


Тема 2.

Поток. Стандартни потоци. Текстови файлове.
Иерархия на потоците. Интерфейс на потоци.
Потоци за вход/изход от файл. Режими на
работа. Флагове на състоянията на потока.
Позициониране във файл.
Пример за функция, която връща големината
на файл.
Пример за функция, която връща брой
редове в текстов файл.

- поток - последователност от байтове, насочени
в определена посока

Иерархия на потоците:



Потоци за изход: (имат put указател)

- форматиран изход:

ostream << <обект>
ofstream - за файлове

- неформатиран изход

- put (char ch)

- поставя 1 символ и отпечата указателя put
нагласно

- write (const char* str, size)

- при двоици файлове

- синхронизация

- flush(); (close() извиква flush())

- позициониране

- tellp() - връща число (позицията на
put указателя)

- seekp(index) - слага put указателя на
даден индекс

- seekp(offset, std::ios::beg/end/curr);

Потоки за ВХОД:

- форматирован ВХОД
istream \Rightarrow <объект>

- Неформатирован ВХОД

- get()
- get(char* buff, Брой символи) \nearrow ^{зете до първото} _{от двете}
- get(char* buff, Брой символи, разделител)
- getline(буфер, размер, разделител)
 \uparrow
по default е \n

- ignore() - игнорира 1 символ
- ignore(Брой символи, разделител)

- позиционироване

- tellg() - индекс на get
- seekg(index) - стага get указателя на индекс
- seekg(offset, direction)
- peek() - връща текущия символ (не мести указателя)

\rightarrow файл - последователност / масив от байтове

Решими на работа

- 1 → ios :: in - отворен поток за вход (четене)
2 → ios :: out - отворен поток за изход (писане)
4 → ios :: ate - слага указателя накрая,
допуска вмъкване на произволни места
8 → ios :: app - слага рпт указателя в края
на файла
16 → ios :: trunc - ако файлът съществува,
съдържанието му се изтрива
32 → ios :: binary - отваря файла в двоичен
режим на работа

→ комбинират се с логическо или

Състояния на поток:



- bad() - показва загуба на информация
- fail() - последната входно/изходна операция е невалидна
- eof() - дали е достигнат края на файла
- good() - ✓ е успешно
- clear() - изчиства състоянието