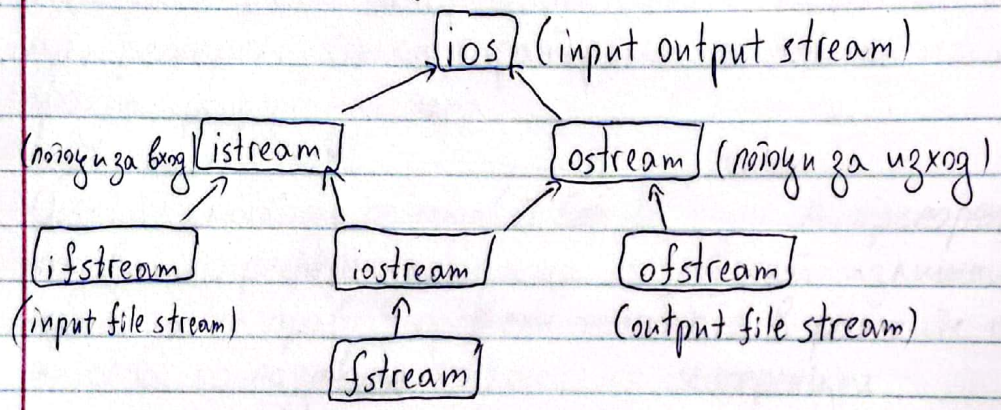
Тема 2

Текстови файлове. Йерархия на потоците. Потоци за вход/изход от файл. Режими на работа. Текущо състояние на поток. Позициониране във файл.

Файловете се делят на два вида:
- текстови файлове - лесни за човека, трудни за програмата
- двоични файлове - трудни за човека, лесни за работа с програмата

Когато говорим за файлове, има термин, който се нарича поток. Потокът представлява това нещо, чрез което програмата комуникира с външни/вътрешни ресурси
В C++ имаме следната йерархия на потоците:



Има 2 вида потоци:
- потоци за вход (istream)
- потоци за изход (ostream)

Има и потоци, които са и за вход, и за изход (iostream)
Пример за потоци са cin - поток за вход и cout - поток за изход. Тези потоци са за комуникация чрез стандартния вход/изход от конзолата.

Вход от файлове.
Отваряне на файл:
ifstream file("име на файл", режим на работа); ← може да се пропусне
Програмата отваря поток, който ще чете от файл. Но потокът може да не успее да отвори файла, ако е повреден или вече отворен. Затова задължително правим проверка дали файлът е успешно отворен:
if (file.is_open())

В края на файла има символ, нареген eof (end of file) и при неговото ? 2
досигане гетенето приключва
Пример за гетене на отделни елементи:
int a, b операторът >> гете до нов ред, eof или ' '.
file >> a >> b;
Пример за гетене на цял ред:
char line[1024];
file.getline(line, 1024);

Нека си представим файла като двулента лента с памет. При гетенето на този файл има указател "get"! Той гете символ по символ, докато не приключи файлът. "getline" придвижва указателя и взима данните от паметта до края на текущия ред.
get-указателя реферира елемента, който ще се прогете при следващата входна операция

Изход от файлове:
Създаване на обекта:
ofstream file ("име на файл", режим на работа);
Ако не съществува файл с такова име, то се създава нов.
Пример за писане във файл.
int a=3, b=10;
file << a << " " << b; Резултат: 3 10

При писане във файлове има put-указател, който реферира мястото, където ще се запише следващия елемент.

Важно е, когато отворим поток, след това рязко да го затворим с file.close()
Операционната система не позволява работа на 2 програми върху един и същ файл и може да възникне грешка, ако не се затвори потока.

Режим на работа:
Режимът е цяло число, 8-битово, като всеки един бит отговаря за различното.
Вместо да се мятат битове, има константи, които могат да се използват
-std::ios::in - отваря файл за гетене (извличане)

- std::ios::out - отваря файл за вмъкване. Допуска се вмъкване на произволно място във файла. Ако файлът съществува, съдържанието му се изтрива.
- std::ios::ate - отваря файл за вмъкване и установява put указателя в края на файла. Допуска вмъкване на произволни места.
- std::ios::app - отваря файл за вмъкване и установява put-указателя в края на файла.
- std::ios::trunc - ако файлът съществува, съдържанието се изтрива
- std::ios::binary - превключва режима от текстов в двоичен.
Тези режими могат да се обединяват с побитово или
Пример:
ofstream outputFile(destName, std::ios::out | std::ios::app);

Флагове на състоянията на потока:
.eof() - дали сме прочели края на файла
.good() - дали всички операции са извършени успешно
.fail() - дали последната операция не се е провалила
.bad() - дали някоя операция не се е провалила
.clear() - изчиства състоянията

Позициониране във файлове:
tellg() - връща позицията на текущия символ в потока за четене (индексът на get указателя)
tellp() - връща позицията на текущия символ в потока за писане (индексът на put-указателя)
seekg(offset, direction) - премества get-указателя на позиция в потока за четене
seekp(offset, direction) - премества put-указателя на позиция в потока за писане
seekg(streampos, idx) - премества get-указателя на позиция idx в потока за четене.
seekp(streampos, idx) - премества put-указателя на позиция idx в потока за писане.

offset - целочислена стойност - отместването от direction
direction - приема следните стойности:
1. std::ios::beg - началото на файла.
2. std::ios::cur - текущата позиция на файла.
3. std::ios::end - края на файла.

(*) std::ios::nocreate - отваря за вмъкване, само ако файлът с указаното име съществува
std::ios::noreplace - отваря за вмъкване, само ако файлът с указаното име не съществува