***Потоци и файлове***

* Малко интуиция какво са потоци

Последователност от елементи, които се обработват точно 1 път.

* Примери за потоци : cin >> , cout <<

статични полета на класовете istream и ostream, не можете сами да си създадете обекти от тези класове

Клиентът използва входни устройства като клавиатура, чрез които записва информацията в поток и после с подходящи класове и оператори тази информация се извлича от програмиста и се обработва.

След края на обработката на тези данни програмистът извежда информацията чрез изходни потоци и клиентът я получава на екрана си.

Други видове потоци – cerr и clog – съобщения за грешки

* Методи на cin и cout

**ostream& put(char)** – записва символа в изходния поток

cout.put(‘A’).put(‘B’).put(‘C’);

**ostream& write(const char\* str, streamsize size)** str – низ; size – брой на символите, които ще бъдат записани;

cout.write("ABCD", 3).write("ABCD", 2);

**istream& get(char& ch)** - извлича един символ и го свързва с променливата ch.

char c1, c2, c3;

cin.get(c1).get(c2).get(c3);

**istream& get(char\* str, streamsize size);**

**istream& get(char\* str, streamsize size, char delim)**

str – низ; size – максимален брой на символите, които ще бъдат извлечени; delim – символ, който се нарича разделител.

char str[10];

cin.get(str, 10, '.'); // abcd.defg.1234

Резултат: str: abcd

char str[10], str2[10];

cin.get(str, 10, '.').get(str2, 10, '?'); // abcd.defg.1234

Резултат: str: abcd str2: .defg.123

**istream& getline(char\* str, streamsize size);**

**istream& getline(char\* str, streamsize size, char delim);**

str – низ; size – максимален брой на символите, които ще бъдат извлечени; delim – символ, който се нарича разделител.

Пример:

char str[10], str2[10];

cin.getline(str, 10, '.').getline(str, 10, '.'); // abcd.defg.1234

Резултат: str: abcd str2: defg

**int peek();**

Връща ASCII кода на поредния символ, но не го извлича от входния поток.

Пример : Когато не знаем дали следващия символ е число или буква и съответно не можем да запаметим директно в променлива int примерно

**istream& putback(char ch);**

Връща символ в потока. Пример:

char c1, c2;

cin.get(c1).get(c2);

cin.putback('9').putback('5');

cin.get(c1).get(c2); cout << c1 << c2;

Резултат: 59

* Състояние на поток :

Задава се чрез множество от битове, които изпълняват ролята на флагове :

-goodbit

-eofbit

-failbit - последната входно/изходна операция е неуспешна.

-badbit - изпълнена е невалидна операция и има загубена информация.

* Методи, с които разбираме състоянието на потока

**- iostate rdstate()** **const** – връща текущото състояние на потока

**void clear(iostate fl = goodbit)//** параметър по подразбиране Модифицира състоянието на потока.

**bool good() const** – връща истина, ако съответния флаг е активиран;

**bool eof() const** – връща истина, ако съответния флаг е активиран;

**bool fail() const** – връща истина, ако съответния е възникнала грешка. Потокът може да се използва, ако флагът се нулира.

**bool bad() const** – връща истина, ако съответния флаг е активиран;

* Работа с файлове

Включване на библиотека за работа с файлове

#include <fstream>

- Декларация

* Отваряне
* Четенe/Писане
* Затваряне

**Put** и **Get** указатели

**Декларация**

**ifstream** iFileName; - за извличане (четене)

**ofstream** oFileName; - за вмъкване (писане)

**fstream** ioFileName; - за извличане и вмъкване

**Отваряне**

fileName.open("file", режим за достъп);

режими за достъп :

**ios::out** – отваряне на файла за писане. Поставя put указателя в началото. Ако файлът съществува то съдържанието му се изтрива.

**Ios::in** – отваряне на файла за четене. Поставя get указателя в началото на файла.

**Ios::app** – Поставя put указателя в края на файла и писането започва от там.

iFileName.open("file", ios::in);

oFileName.open("file", ios::out);

ioFileName.open("file", ios::in|ios::out);

**Четене и писане**

ifstream fin("file1", ios::in);

ofstream fout("file2", ios::out);

while (fin.get(ch)) fout << ch;

while (fin.get(ch)) fout.put(ch);

Копира дословно файла.

while (fin >> ch)) fout << ch; Не копира интервали, нов ред, табулация.

Функции :

**istream& seekg(streamoff p, ios::seekdir r)** – премества get указателя, с р байта относно режима на позициониране r (ios::beg,ios::end,ios::cur)

**ostream& seekp(streamoff p, ios::seekdir r)** – премества put указателя, с р байта относно относно режима на позициониране r (ios::beg,ios::end,ios::cur)

**streampos tellg()** – връща текущата позиция на get указателя на файла.

**streampos tellp()** – връща текущата позиция на put указателя на файла.

Пример:

file.seekg(0, ios::end);

long loc = file.tellg();