Уровни языков:

1. Машинные коды
2. Операции процессора
3. Ассемблер (Windows), C/Shell, Bash (Unix)

**2.5) C++, Fortran**

1. Java, Python
2. JavaScript, HTML

Написание кода -> Проверка синтаксиса -> Лексический анализ -> Синтаксический анализ -> Сборка

*#include <iostream> // директива препроцессора  
using namespace std; // используемое пространство имен  
int main() {  
 cout <<”WTH!”;  
 return 0;  
//функция целочисленного вида, принимает пустой набор аргументов, возвращает нулевой результат  
}*

Операторы:  
1. ; - пустой оператор (разделитель)  
2. {} – составной оператор (объединяющий)  
3. = - оператор присваивания  
4. int, bool, double – типы данных, ключевые слова  
5. break, continue – «опасные» операторы

(!) Переменные, созданные внутри составного оператора, существуют только внутри этого оператора

Функция f принимает два аргумента и вычисляет НОД:

*int f (int a, int b) {  
if (b!=0)  
return f(b, a%b)  
else   
return a;  
}*

Факториал числа:

*int a(int n) {  
int s=1;  
for (int i=1; i<=n; i++)  
s=s\*I;  
return s;  
}*

Рекурсивное вычисление факториала:

*int a(int n) {  
if (n==1)   
return 1;  
return n\*a(n-1);  
}*

s\*=i более быстрый вариант, чем s=s\*i, поскольку не создает новых ячеек памяти.  
В свою очередь, s++ - процессорное увеличение на 1, САМЫЙ быстрый способ из всех