```
Diff b2n Lang. VS program Lang

H2H => Language
H2C => Program Language  C C++
   I
Procedural Language

//32 Keywords
// "Compiler" and Interpreter
// Operators,  Decision, Loop, (iterative) Switch, Function Pointers Array, String FILE ..... [END OF C]

// ..............................SOURCE CODE ..................
#include <stdio.h>

int main()
{
    printf("%f",c);
    return 0;
}
// ..............................

//Processing
 Source Code (.c) => "PREPROCESSOR" =>

// ..............................  EXTENDED SOURCE CODE ..................
int main()
{
    printf("%f",c);
    return 0;
}
// ..............................

```

```c
#include <stdio.h>

int main()
{
    int x, y, z;
    printf("Enter two Numbers");
    scanf("%d %d", &x, &y);
    z=x+y;
    printf("Sum of %d and %d is %d",x,y,z);
    return 0;
}



// 3 basic Data types int (%d) 6  9  2  3     float (%f) 4.5 3.9      Char(%c)   'a' 'b' 'c'
```

input

```
Enter two Numbers4
6
Sum of 4 and 6 is 10

...Program finished with exit code 0
Press ENTER to exit console.
```

```c
#include <stdio.h>

int main()
{
    float x, y, z;
    printf("Enter two Numbers");
    scanf("%f %f", &x, &y);
    z=x+y;
    printf("Sum of %f and %f is %f",x,y,z);
    return 0;
}



// 3 basic Data types int (%d) 6  9  2  3     float (%f) 4.5 3.9      Char(%c)   'a' 'b' 'c'
```

input

```
Enter two Numbers2.5
3.7
Sum of 2.500000 and 3.700000 is 6.200000

...Program finished with exit code 0
Press ENTER to exit console.
```

```
//Processing
Source Code (.C) => "PREPROCESSOR" => Extended Source Code (.S) => "COMPILER" => Assembly Code (.ASM) => "ASSEMBLER"

=> Object File (.obj) of Source Code => "LINKER" => A single file without any extension => "LOADER" => Output File (.exe)


//Source      Program      Process
//Code

//HDD         RAM          PROCESSOR
```

```
// 3 basic Data types int (%d) 6  9  2  3     float (%f) 4.5 3.9     Char(%c)   'a' 'b' 'c'


x Value  2.5  (x)
memory address 1000  (&x)


// Operators
1.  Arithmatic Operators
+ - * / % etc..
    / and % float DType will not work
    -5%2 = -1   but  5%-2 = 1   (Sign depends on the numerator)


2.  Relational Operators will return
    NONZERO or ZERO
    YES or NO
    TRUE or FALSE

    >  >=  <  <=  ==  !=

    int a=3, b=2, c=2;
    d= a>b;
    printf("%d", d);   // OUTPUT 1
```

```
30      TRUE(1) or FALSE(0)
31
32      >  >=  <  <=  ==  !=
33
34      int a=3, b=2, c=2;
35      d= a>b;
36      printf("%d", d);    // OUTPUT 1
37      d= a<b;
38      printf("%d", d);    // OUTPUT 0
39      d= a==b;
40      printf("%d", d);    // OUTPUT 0
41      d= a!=b;
42      printf("%d", d);    // OUTPUT 1
43
44
```

```
3. Assignment Operator
   a=3

4. Logical Operators will return
    NONZERO or ZERO
    YES or NO
    TRUE(1) or FALSE(0)

   &&  Logical AND    (It will return 0 if anyone of the INPUT is 0)
   ||  Logical OR     (It will return 1 if anyone of the INPUT is 1)
   !   Logical NOT    (will reverse the input as output)


   int a=3 (Nonzero), b=-2(Nonzero), c=0 (zero);
    d= a && b && c;      ZERO
    d= a || b || c;      ONE (only considering the value of a)
    d= a || b && c;      ONE (only considering the value of a)
    d= c && a || b;      ZERO
    d= a && c || b;      ONE
```