SKILL MILL

Programming Languages and Translators

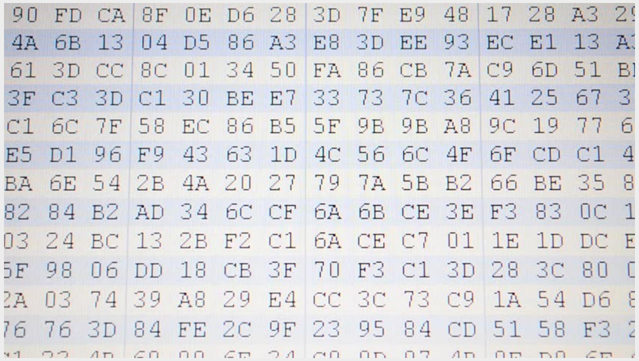
Program

Some Languages

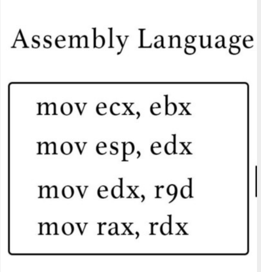
python™
JS
5
HTML
5
CSS
3
C++
TS
R
THE SCHEME PROGRAMMING LANGUAGE
Java
Kotlin
C#
Perl
php
Scala
Swift
MATLAB SIMULINK
Microsoft SQL Server
R
GO

Low level languages

- Make special use of hardware and machine-dependent instructions.
- Closer to machine code
- Doesn’t take up as much space in memory
- Runs more quickly



```
 90 FD CA 8F 0E D6 28 3D 7F E9 48 17 28 A3 2
 4A 6B 13 04 D5 86 A3 E8 3D EE 93 EC E1 13 A
 61 3D CC 8C 01 34 50 FA 86 CB 7A C9 6D 51 B
 3F C3 3D C1 30 BE E7 33 73 7C 36 41 25 67 3
 C1 6C 7F 58 EC 86 B5 5F 9B 9B A8 9C 19 77 6
 E5 D1 96 F9 43 63 1D 4C 56 6C 4F 6F CD C1 4
 BA 6E 54 2B 4A 20 27 79 7A 5B B2 66 BE 35 8
 82 84 B2 AD 34 6C CF 6A 6B CE 3E F3 83 0C 1
 03 24 BC 13 2B F2 C1 6A CE C7 01 1E 1D DC E
 5F 98 06 DD 18 CB 3F 70 F3 C1 3D 28 3C 80 0
 2A 03 74 39 A8 29 E4 CC 3C 73 C9 1A 54 D6 8
 76 76 3D 84 FE 2C 9F 23 95 84 CD 51 58 F3 2
```



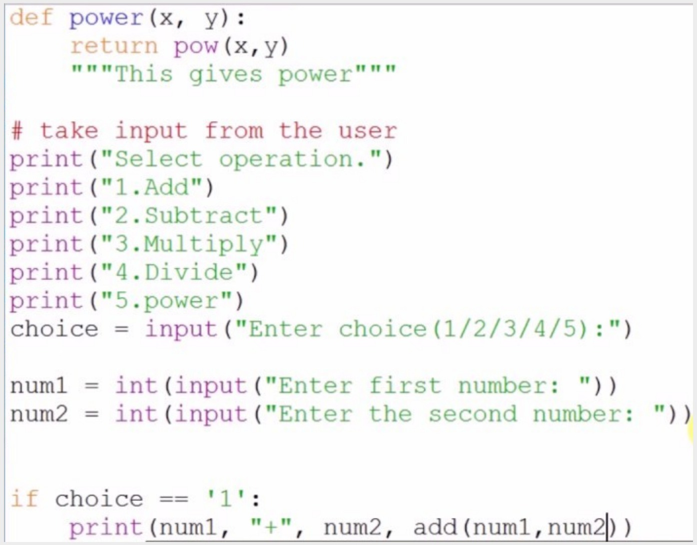
Assembly Language

```
mov ecx, ebx
mov esp, edx
mov edx, r9d
mov rax, rdx
```

```
11110011111000111011111110001111001011111000001001000111
1000111100011100101111110001110000110001101110010011011
110001111100111111110111111000001011100000000000001000000
0011110111101111111111110001111111111111100000000000010011
110111011000000010011001110111111111111011001101001100011
11100000111011111101110111110111101100001011101110000011
11100010010011111000100011000011000010111011101111100000
11110011001011100111111111111111111000001110111110111110
0111110000100001010111111111100011101111110111011111111111
01110000100111100100001100000000010000011001111110111110
11101010010000001011111101111110001101111111111111110111
1110111110000001101001011110001011111110111111011001111
1110111101001111111100111111001010111110111111110111111
1100111111111000110101110010110011000010111001001111111
1100111101111111000100101101111100000100110011001110011
1011100000110001111111111111111101000011010001000011111
1110110011011100100111110101011011111100111011111111111
000000111111111111011111111111111110110011000110110111011
0111111011111111111011100111111111111111111000111001111111
00011111111111101000100001101111111111001100000000111100
```

High level languages

- High level languages abstract the low level implementation details
- It is easier to understand since it is closer to human language
- takes less time to write programs
- Debugging or finding errors in the code is easier
- Code takes longer to run than code of low level languages

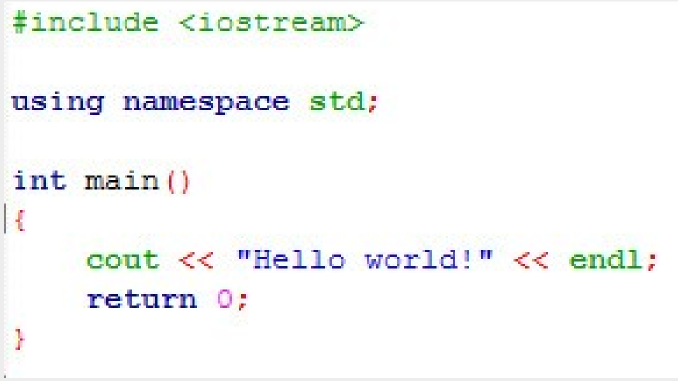


```python
def power(x, y):
    return pow(x,y)
    """This gives power"""

# take input from the user
print("Select operation.")
print("1.Add")
print("2.Subtract")
print("3.Multiply")
print("4.Divide")
print("5.power")
choice = input("Enter choice(1/2/3/4/5):")

num1 = int(input("Enter first number: "))
num2 = int(input("Enter the second number: "))


if choice == '1':
    print(num1, "+", num2, add(num1,num2))
```



```cpp
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!" << endl;
    return 0;
}
```

Translators

Compiler

- Translates a whole program from a high-level language to machine code in one go
- An executable file of machine code is produced
- Once compiled, the program doesn’t need to be recompiled, this means that compiled programs can be used without the compiler
- One high-level language translated into several machine code instructions

Interpreter

- Executes a high-level language program a statement at a time
- No executable file of machine code produced
- Interpreted programs can’t be used without the interpreter
- One high-level language program statement may require several machine code instructions to be executed
- Used when program is being developed

Assembler

- Translates a low-level language program into machine code
- An executable file of machine code is produced
- Once assembled, the program doesn’t need to be reassembled, this means that the assembled programs can be used without the assembler
- One low-level language is translated into one machine code instruction