

1. If we only use one language in all programming domain, the language we used must be very heavy. It may have too large standard library, too many special word, even too hard to learn because it has too many feature. Moreover, I think it is hard for a language to have all feature. Take generic as example, In Java, all variable(except primitive type) are object, so it's generic system must use erasure to implement. But in C++, we can use compile-time type checking template which make C++ generic powerful and more flexible. Therefore, if a language has structural type system, it has limited generic and otherwise it may has powerful generic.
2. Generally, a efficient program means it is hard to read or hard to modify. If the hardware is inexpensive, I think we have to write program more structural and more readable. But we also have to try our best to reduce unnecessary preformance loss which has only raise a little readability like the above code.

```
HashMap<> classForScore = new HashMap<int, char>() {{  
    put(0, 'F');  
    put(1, 'F');  
    put(2, 'F');  
    put(3, 'F');  
    put(4, 'F');  
    put(5, 'F');  
    put(6, 'D');  
    put(7, 'C');  
    put(8, 'B');  
    put(9, 'A');  
}};
```

```
// get class of student's grade
```

```
// Higher Readability one
```

```
char getClassFromGrade(int grade){  
    return classForScore.get(grade % 10);  
}
```

```
// Higher Performance one
```

```
// I think this is better
```

```
char getClassFromGrade(int grade){
```

```
if (grade >= 90)
    return 'A';
else if (grade >= 80)
    return 'B';
else if (grade >= 70)
    return 'C';
else if (grade >= 60)
    return 'D';
else
    return 'F'

}
```