**对C语言的产生发展、标准化历程、语言特征的研究报告**

——CS1904 王翰辉

一、问题的提出：

基于C程序设计学习的需要，有必要在正式开始编写程序之前了解这门程序设计语言究竟是什么，来自哪里(历史沿革)，将要去向何方(我们可以用它干点什么)。

二、调查方法：

查阅资料——《C语言程序设计》、《C Primer Plus》以及老师的PPT

三、研究结果：

1. C语言的产生发展

C语言是一门程序设计语言，那么究竟什么是程序设计语言呢？可以认为：这是一种由一些符号构成，被用于定义、组织并完成各种各样的计算任务的指令体系。或者简单认为，程序设计语言是我们指导“愚蠢的”计算机去完成一系列任务的工具。这些语言可以分为：机器语言，汇编语言以及高级语言。机器语言，顾名思义就是机器真正能够理解使用的语言，也就是一连串的0、1组合。虽然机器能够轻而易举地理解这种“二进制语言”，但人类则更习惯与符号打交道，于是出现了汇编语言，将机器语言的数字串改成了一系列助记符。但是汇编语言仍然是面向机器的，只是一种符号化的机器语言，并不能被大部分“人类”使用。于是乎高级语言就此登上了历史舞台，从此这种面向用户，面向应用的语言一发不可收拾，在上个世纪60年代，C语言“最早的起源鼻祖”FORTRAN语言出现了，它是 Formula Translation的缩写，尤其名字可知它的程序的编写应当已经比较接近人类使用的自然语言并能够使用数学公式了。而很有意义的是它编译后产生的目标代码的执行速度极为可观，大大提高了高级语言的代码执行效率。然而这仍然不是能让一般人员使用操作的语言，于是一种设计通用语言的念头出现了。在这一背景下，接连出现了ALHGOL、CPL、BCPL等语言。70年代，Ken Thompson通过对BCPL语言进行简化改进，设计了B语言，引入了今天为我们所熟知的“++”、“--”等操作。随后在B的基础上，Dennis M. Ritchie设计出了拥有数据类型的C语言。此时的C是基于UNIX操作系统的，如今可被理解为传统的C。1978年以后C开始被各种类型的计算机使用，从UNIX系统中独立出来了。

2. C语言漫长的标准化历程

C语言产生至今已经走过了超过40年的旅程，在其中历经了多次大大小小的“改造”，以使它能够被更为广泛的应用。

—1989年 此时的美国人已经意识到C语言的普遍化是不可逆转的时代潮流，因而“未雨绸缪”，制定了第一个C(国家)语言标准——ANSI 89。

—1990年 国际标准化组织ISO接受了美国标准，将其称为标准的C，也即是C90。

—时间来到了90年代以后，随着计算机的不断发展，人们寄希望于利用计算机处理更大范围，更大精度的数据计算，此地标准的C渐渐不能满足人们日益增长的需求。于是ISO/IEC95年开始征集对C继续改进规范的意见。

—1999年 时隔四年，这项浩大的工程结束了，引进了更多数据类型，更大数据范围，更好计算精度的C99正式登上了历史舞台。

—2011年 新世纪的第一套C语言国际标准出台，史称C11。

可以看出在刚刚出现的时候，对于C语言的使用标准是存在许多争议的，因而在其产生之初不断有新的标准出台，而随着计算机世界的快速发展，C语言的使用越来越普遍，其标准也逐渐为世人所广泛接受，改动的频率与幅度逐渐下降。我们有理由相信我们如今所学习的C语言将在适用于我们一生往后的学习生活。

3. C语言的特征

在此基于篇幅考虑，某些特点只大致给出个人理解。

(1) 简洁紧凑

关键字集(包含数据类型，声明、循环、条件的基本语句等)少，共37个元素。以之构造的程序基本语句总共只有12种。好处：语句类型少，方便记忆。(我认为的坏处：编写复杂程序时加大了工作量)。

(2) 目标代码质量高

简单来说就是占用的储存空间小，代码执行效率高。这是往后如Python

等语言所无法比拟的(尽管它们让我们敲的代码减少了许多)。

(3) 语言表达能力强

以种语言的表达能力取决于它的类型集、操作集、语句集

① 类型集

独到之处——指针。可以视为掌握一切变量地址(它们储存在哪里)从而

能够进行快速间接访问的手段。

② 操作集

操作集包括运算符，库函数的操作符等。对C而言，它们数量大并且语义接近汇编语言，从而提高了代码质量。

③ 语句集

由前已知十分简洁，而在满足应用需求的前提下，语言集越简洁越好。

(4) 流程控制结构化

在高中我们就已经初步学习了顺序、分支、循环三种基本结构。抛开顺序不谈，分支结构在C语言中可以用if-else语句或switch-case语句实现(前者有两个基本分支，后者可以拥有多个),而while、do-while、for语句则能够满足多种循环形式需求(三者有区别但可以设法做到互换)。而循环和switch-case语句中break或continue都可以起到一定程度上提高效率的目的(前者打破循环，后者快进到下一轮循环中)。还有那个破坏结构调理性但是适当使用能够提高效率的goto语句。

(5) **弱语言类型**

在此给出书本定义和我自己的理解，因此会包括一些相关的其他内容。

① 动态类型语言

在运行期间才会去做数据类型检查的语句，不用给变量指定类型，在你给它第一次赋值的时候会确定它的数据类型。

② 静态类型语言

在定义变量时就要给定它的数据类型。

③ 强类型语言

一旦明确了数据类型就不能进行更改(除非进行强制类型转换)。

④ 无类型语言

在进行赋值操作时根据右操作数类型推导并转换左操作数类型。

⑤ 弱类型语言

在进行赋值操作时根据左操作数类型要经过适当转换向右操作数类型看齐。

注意：动静区别在于定义变量时要不要进行数据类型的指定，而强弱无则是在进行赋值操作时加以区分的，两种类别之间没有等价联系，如Python是动态语言，但同时也是强语言，而C是静态语言，同时也是弱语言。

为什么C是弱语言？

在这里我们定义a是一个double型变量，然后我们让a = 6/5，并且输出6/5和a。我们会发现前者输出1，后者输出1.000000。首先因为int/int = int，所以6/5就是等于1，而不是1.2。但是由于C语言是弱类型语言，a = 6/5将会使右边的操作向左操作数看齐，进而将1转换为1.000000。弱类型语言不会影响右操作数本身是多少，但是会在运算结束后向左操作数看齐。

(6) “中级语言”特性

简而言之C语言介于低级和高级之间，既有高级语言的表达能力，接近自然语言和数学语言的特性，同时还具备低级语言对硬件系统良好的操纵与控制能力。

(7) 书写使用自由灵活

语句以“;”结尾，可以自如地在编程时进行换行。

(8) 可移植性好

一段C语言代码稍加改动甚至不经改动便可以从一种型号的计算机转移到另一种型号的计算机上，经过重新编译链接后运行。

四、总结及注：

这是基于自身所学及课本所述的研究报告，内含我个人的一些不成熟的观点，而我也在其中对什么是弱语言以及为什么C是弱语言进行了解释，可见于下划线及加粗部分。