## 第三部分 程序语言及程序设计基础

## 知识点

1. 程序设计语言(机器、符号、高级语言)的特点
2. 标识符
3. 数据类型及数据类型的三要素（逻辑结构/表示、存储结构、数据操作）
4. 存储空间，存储地址、存储单元，位(bit (b))与字节(Byte (B) )，存储容量单位(KB、MB、GB等)
5. 变量及变量的三要素
6. 三类常量：文字常量、命名常量、符号常量；命名常量和符号常量的区别。
7. 表达式，表达式的递归形式定义
8. 三种基本语句：赋值、输入、输出
9. 三种基本程序结构：顺序、分支、循环

## 第四部分 算法设计方法

## 知识点

1. 程序设计的步骤？
2. 什么是算法？算法的五大特征
3. 结构化编程，自顶向下、逐步求精的方法
4. 算法的描述方法（N-S流程图）
5. 迭代算法、穷举算法
6. 算法思路：问题抽象（数学建模），求解问题的步骤

## 第五部分 子程序（函数）

## 知识点

1. 函数的定义、函数原型
2. 函数的调用
   1. 函数的参数原理，形参与实参
   2. 函数的调用过程：堆栈、函数活动记录
3. 子程序设计（函数设计）

高内聚/低耦合，参数设计

减少代码冗余

4、十进制数与N进制数(2、8、16等)的“相互转换”

5、变量的作用域(局部变量、全局变量)

## 第六部分 递归（函数递归）

## 知识点

1. 递归的概念，递归函数定义
2. 递归过程，基于函数调用过程能够自主分析递归过程，得出结果。
3. 递归算法的分支结构，包括哪些分支？
4. 递归程序设计，化简为同类问题，分解直至能求解
5. 递归函数的参数设计，不建议使用全局变量。

## 第七部分 数组

## 知识点：

1. 数组的概念：逻辑结构、存储结构
2. 数组的定义、下标运算符
3. 数组的逐元素访问
4. 数组作为函数参数：1. 数组元素作为参数（值传递）；2.数组名作为参数（相当于引用传递）
5. 字符串与字符数组： 存储特征 --结束符(‘\0’)；整体的输入与输出操作；字符串操作函数：拷贝、比较等
6. 二维数组： 特别是数组名作为函数参数的用法

## 第八部分 指针与数组

## 知识点：

1. 指针的概念，指针的定义（语法）
2. 指针自身的类型和指针所指向数据的类型
3. 指针运算符： 间接访问\* 取地址 &。
4. 指针作为函数参数：相当于引用传递的效果
5. 指针变量指向数组，指针的算术运算、关系运算

注意：ptr+=i 与 ptr+i 的区别

1. 指针与数组：
   1. 数组元素的4种访问方式（下标、偏移量）
   2. 指针与数组： 动态数组，堆区分配存储
   3. 无类型指针void \* 的含义，对它能够进行哪些运算
   4. 指针数组，指向指针的指针，双重间接访问(\*\*p)
   5. 动态分配多个字符串的处理（分配顺序：先分配指向字符串的指针空间，再分配字符串空间；释放顺序：先释放字符串空间，再释放指向字符串的指针空间）。以及在此基础上对多个字符串的操作，如排序、子串等。
   6. 通过指针访问和处理二维数组。ａ［ｉ］［ｊ］＝　＊（＊（ａ＋ｉ）＋ｊ）

## 第九部分 自定义数据类型――结构体和枚举

## 知识点：

1. 再论数据类型（三要素），用户自定义（构造）数据类型
2. 结构的定义，结构成员的访问，结构变量的操作（结构体的整体赋值）
3. 结构作为函数参数 结构传值、传地址
4. 枚举的定义

## 第十部分： 数据结构+算法=程序

## 知识点：

1. 基于数组、结构体等，对现实生活中较复杂问题进行信息与数据抽象，以及数据结构设计（如一个班学生的选课信息、一个图书馆的书目列表等），会使用typedef定义相应的数据类型(结构)。
2. 在数据结构设计的基础上进行算法设计
3. 基本算法设计思想：迭代、穷举、递归