程序的三大结构

1. 顺序结构
2. 选择结构
3. 循环结构

顺序结构：程序按照自上而下，依次执行。

选择结构：根据条件判定，确定选择执行哪个一语句或者语句块

循环结构：根据条件判定，反复执行语句或者语句块

流程图：根据程序的执行顺序，画出的程序流程图解。

图例：表示某一现实生活中物体的代替品。

程序开始和结束 圆角矩形

程序输入和输出 平行四边形

程序执行语句 矩形

程序条件判断 菱形

带有方向的直线 流程线

程序的连接点

程序的灵魂=算法+数据结构

程序开发思路：

分析：1、判断一个三位是否是素数。

2、水仙花

组成三位数的每个位上的数据的立方之和等于该数据本身。

3、闰年

一个年份能被4整除并且被100整或者400整除，说明是闰年

画图：

顺序结构

开始

结束

选择结构：

1. 单分支选择

语法：if(条件表达式)

{

语句或者语句块

}

原理：如果条件表达式成立，执行其身后的一条语句或者语句块。

1. 双分支选择

语法：if(条件表达式)

{

语句1；

}

else

{

语句2；

}

原理：如果条件表达式成立，则执行语句1，否则执行语句2；

1. 多分支选择

语法： if(条件表达式1)

{

语句1；

}else if(条件表达式2)

{

语句2；

}else if(条件表达式3)

{

语句3；

}else

{

语句4；

}

原理：如果条件表达式1成立则执行语句1，否则判断条件表达式2，如果成立则执行语句2，否则继续判断条件表达式3，如果成立则执行语句3，否则执行语句4；

推理：有n 个条件,判断执行n+1条语句。

高中会考： A B C D

销售人员： 提成 个人所得税（）

零件的成品率：

开关语句

Switch(表达式){

Case ‘对比参数’: 语句1； break;

Case ‘2’:语句2；break;

Case ‘3’:语句3；break;

Default:语句n;break;

}

原理：

如果表达式与其中的一个对比成功，则执行其身后的语句。如果该语句身后有break，则退出该语句体。否则直到碰到第一个break语句的时候才退出该语句体。如果对比不成功，则执行default语句。

月份的天数

1 3 5 7 8 10 12 31

2 28 29

4 6 9 11 30