学习方法

1. 建立逻辑思维
   1. 多分析
   2. 理解逻辑算法
   3. 多练、多敲、多读
2. 编程习惯
3. 大胆动手
4. 保持好奇好
5. 乐观面对错误

就业能力

1. 分析能力
2. 独立思考能力
3. 程序阅读能力
4. 程序排错能力
5. 独立编写程序能力
6. 团队协作能力
7. 学习能力
8. 善于总结的能力

专业技能

逻辑分支

1. 程序的单分支结构IF语句
   1. 顺序结构：程序是按顺序从上往下执行的。
   2. 选择结构
      1. 概念：选择结构用于判断给定的条件，根据判断的结果判断某些条件，根据判断的结果来控制程序的流程。
      2. ？： ：三元运算符
      3. If语句
         1. If(){} :单分支选择语句
            1. 格式：  
               If(表达式){  
                语句组;  
               }  
               注：如果if中条件成立时，后面只有一条语句，大括号可以省略，但是如果大于1条语句时，必须加大括号。（建议无论多少条语句，都要加上大括号（潜规则））
            2. 逻辑关系：当程序执行到if时，先计算表达式的值，如果值为true，则执行后面大括号中的语句；如果值为false时，执行if语句后面的其它语句
            3. Demo:

根据成绩判断是否发放清华大学的通知书？

输入任意两个数，然后交换位置输出（如：a=4,b=5输出a=5,b=4）

输入任意三个数，由大到小输出

* + - 1. If(){}else{} :双分支选择语句（多分支选择语句）
         1. 格式：  
            if(表达式){  
             语句组A;  
            }else{  
             语句组B;  
            }
         2. 逻辑关系：  
            当程序执行到if时，先计算表达式的值，值为true时：执行后面语句组A；值为false时，执行else后的语句组B。
         3. 注：能用三元解决问题的，优先使用三元，否则使用if
         4. Demo

求两个数中的最大值？

求三个数的最大值?

输入一个成绩，判断是升班还是重修

* + - 1. If(){}else if(){}…else if(){}else{} :多分支选择语句
         1. 格式：  
            if(表达式1){

语句组1;

}else if(表达式2){

语句组2;

} ……

else if(表达式n){

语句组n;

}[else {

语句组n+1;

}]  
 注：写中括号的地方是可以省略的语句

* + - * 1. 逻辑关系：当程序执行到if时，先计算表达式1的值，值为true时，执行语句组1；值为false时，再计算表达式2的值，值为true时，执行语句组2；值为false时，再计算表达式3的值，依此类推，直到计算表达式n的值，值为true时，执行语句组n,值为false时，执行语句组n+1;
        2. Demo:

任意输入一个数字，判断是星期几？

判断成绩优(90-100)良(80-89)中(70-79)差(60-69)不及格(<60)

设计一个具有+、-、\*、/、%的简单计算器

第二节：应用

1. 判断一个整数，属于哪个范围：大于0；小于0；等于0
2. 判断一个整数是偶数还是奇数，并输出判断结果
3. 开发一款软件，根据公式（身高-108）\*2=体重，可以有10斤左右的浮动。来观察测试者体重是否合适

第三节

1. 程序的多分支结构switch语句、break语句
   1. Switch开关、多分支选择语句
      1. 格式：  
         switch(表达式){  
          case 常量表达式1 : 语句组1;[break;]  
          case 常量表达式2 : 语句组2;[break;]  
          ……  
          case 常量表达式n : 语句组n;[break;]  
          [default : 语句组n+1;[break;]]  
         }  
         [注]:  
         1.加上break时，default与case 的顺序可以随意  
         2.break与default可以省略  
         3.不允许出现相同的case值
      2. 逻辑关系：  
         当程序执行到switch时，先计算表达式的值，如果该值与下面某一个case后的常量表达式的值一致时，执行该case后的语句组；如果语句组后有break语句时，则终止switch语句，退出switch语句；如果语句组后没有break语句时，则继续执行其后的所有语句组，直到遇到break或右大括号为止。
      3. Demo:
         1. 任意输入一个数字，判断是星期几？
         2. 输入一个0-6的整数，判断哪一天是工作日，哪一天是休息日？
         3. 输入一个月份，输出这个月有多少天？
         4. 判断成绩优良中差不 及格
         5. 计算某日是该年的第几天？
         6. 用switch实现简单计算器
         7. 根据血型与性别测试性格
         8. 设计一个简单的猜拳游戏

第四节：应用

1. 案例：输入分数，判定成绩等级（if和switch两种写法）
2. 输入月份，显示当月的天数  
   要求：利用case穿透简化代码

第五节：综合应用

1. 制作一个表单
   1. 要求：
      1. 点击保存按钮时，使用JS判断用户的联系电话是否为纯数字，如果用户输入有错误，使用alert()弹框警告
      2. 判断用户名是否填写，如果空白，使用alert弹框警告