Lua note 1

Now my work content primarily involved circuit automation test and I am responsible to the software part. And it is also the first time contact to script language Lua.

--[[

1.An int array a[10]= {3,4,6,7,8,4,2,1,9,10},sort it from small to large, then delete the number which appear more than once.

2.the length of array less than 10 or equal to 10,each number in the array is between 0 and 9, if 0 can represent each number, judge the number in the array is continuous.

3.write a method to come true BitOr ,BitAnd in Lua.

BitOr   (<=20 lines)

BitAnd  (<=20 lines)]]

        day={3,4,6,7,8,4,2,1,9,10}    --the array

        days={}

        for i=1,10  do

            for j=1,10 do

                if  day[i]<day[j]

               then day[i],day[j]=day[j],day[i]

            end

            j=j+1

        end i=i+1

    end                    --bubble sort。

    l=1

    for m=1,10 do

        if(day[m]~=day[m+1])

            then days[l]=day[m]    -- use the table “days” to represent table “day”

                l=l+1

            else days[l]=day[m+1]

            end

            end                --if the next number is equal to itself,then the key won’t count.

    for i,v in ipairs(days) do

        print(v)

    end

    --2

    function istableorder(tbtest  )

    i=#tbtest x=1 k=0;

    for j=1,i-1 do

        if(tbtest[j]-tbtest[j+1]==-1)  then --judge whether the value the number sub the next number is equal to -1

             x=j+1

        elseif (tbtest[j]==0)  then      -- if is not equal to -1 then judge whether the first number is equal to 0

             x=j+1

             k=k+1

             z=j+1 -- count the number of 0

        elseif (tbtest[j+1]==0) then

               x=j+1

        else

            return(false)

        end

    end

    --print(k)

    --print(x)

    --print(z)

    if(x==#tbtest)then

        if (z-k==1) then

          return(true)

        elseif  (tbtest[z]-tbtest[z-k-1]~=k+1) then

        return(false)

        else return(true)

        end

    else return(false)

    end

end

y={0,0,0,0,0,6}

print(istableorder(y))

--bitor bitand

function bitand(num1,num2,bit )     --BitAnd

n={}

a=bit

val=0

for i=a,1,-1 do

if num1>(2^(i-1)-1) then       --（2^(i-1)-1）change it to binary number if larger than （2^(i-1)-1）,then the i is equal to 1,else it is 0

   if num2>(2^(i-1)-1)  then

      n[i]=1

      num1=num1-2^(i-1)

      num2=num2-2^(i-1)

   else n[i]=0                 --And calculate.

        num1=num1-2^(i-1)

   end

   else n[i]=0

      if num2>(2^(i-1)-1) then

         num2=num2-2^(i-1)

      end

end

end

for i=#n,1,-1 do

val=val+n[i]\*2^(i-1)           --return to decimal.

end

return(val)

end

print(bitand ( 16,8,4 ))

function bitor( num1,num2,bit )

n={}

a=bit

val=0

for i=a,1,-1 do

    if(num1>(2^(i-1)-1)) then

        n[i]=1

        num1=num1-2^(i-1)

        if(num2>(2^(i-1)-1)) then

          num2=num2-2^(i-1)

        end

    elseif(num2>(2^(i-1)-1)) then

        n[i]=1

        num2=num2-2^(i-1)

    else n[i]=0

    end

val=val+n[i]\*2^(i-1)

end -- body

return(val)

end

print(bitor(18,8,5))

It is the first time to practice Lua.