



Network Thinking

Web Programming

Web编程，个人作品的动态网页编程

zxu@ict.ac.cn
zhangjialin@ict.ac.cn

Outline

- What is network thinking?
- Network terms
- Connectivity
 - Naming
 - Topology
- Protocol stack
 - The Web over TCP/IP stack
 - Web programming 个人作品：通过设计动态网页实现创造性表达
- Network laws
 - Performance metrics
 - Network effect
- Responsible computing

These slides acknowledge sources for additional data not cited in the textbook

国科大同学的Peers: 全球英才

- 哈佛大学计算机科学导论课程

This is **CS50**

Harvard College

Spring 2021

“11 weeks, 10-20 hours per week”

Week 0 Scratch

Week 1 C

Week 2 Arrays

Week 3 Algorithms

Week 4 Memory

Week 5 Data Structures

Week 6 Python

Week 7 SQL

Week 8 HTML, CSS, JavaScript

Week 9 Flask

Week 10 Ethics

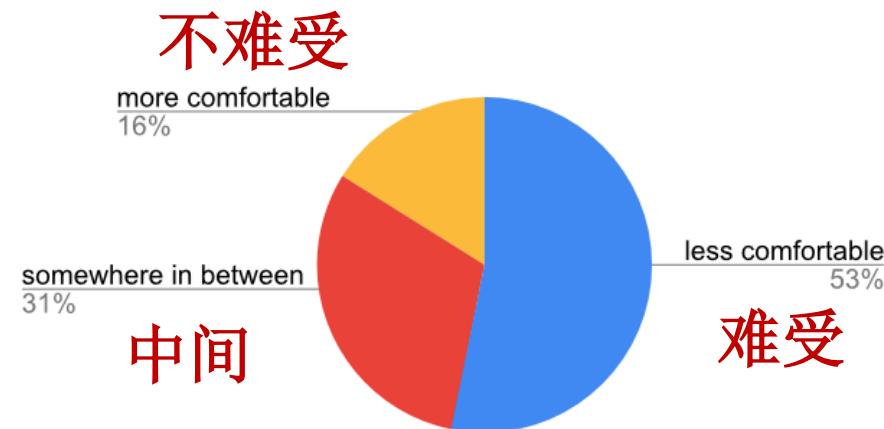
700名学生上大课

+
助教辅助自学

Accordingly, plan to

- watch lectures on Mondays,
- submit quizzes by Tuesdays,
- attend class on Tuesdays,
- submit labs by Thursdays,
- optionally attend tutorials on Wednesdays, Thursdays, Fridays, Saturdays, and/or Sundays, and
- submit problem sets by Sundays.

与其他同学相比，
只有我觉得课程难吗？



大多数同学每周花10小时做作业
10+ hours per week on problem sets

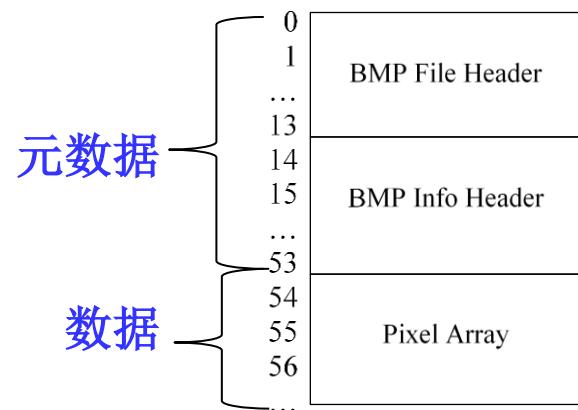
英国的两个例子

- 剑桥大学为什么发明“树莓派”电脑 Raspberry-Pi
- 英国教育部中小学计算机教育指南
 - 每个学生 (pupil) 应该学习两个编程语言，
其中一个是文本类编程语言 (one of which is textual)

信息隐藏实验的三层收获

- **基本要求：** 隐藏 HungLouMeng.txt 到 Autumn.bmp

- hide文本到图像；从图像show文本



- **掌握原理：** 定位到合适位置并操作该位置数据（了解CS的一个基本原理：定位机制）

- 理解文本文件和图像文件的存储**格式**，即哪些字节（byte）存储数据、元数据
 - .txt文件每个字节是一个**ASCII字符**；.bmp文件采用bitmap格式，Pixel Array不从0地址开始
 - 将两个**文件读入到**两个**变量**，变量的**数据类型**都是**字节切片**（byte slice）
 - 用**for**循环将文本文件的每一个字符隐藏到图像文件Pixel Array的多个像素
 - 每次迭代（iteration）做定位并操作，有两个要点
 - **定位**到（字节切片的）合适位置（基本机制仍是第二章的 **base + index*k + offset**）

基址 + 索引*k + 偏移量

文本文件的字符 $t[i]$ is hidden in

$p[86+(i*4)]$, $p[86+(i*4)+1]$, $p[86+(i*4)+2]$, $p[86+(i*4)+3]$

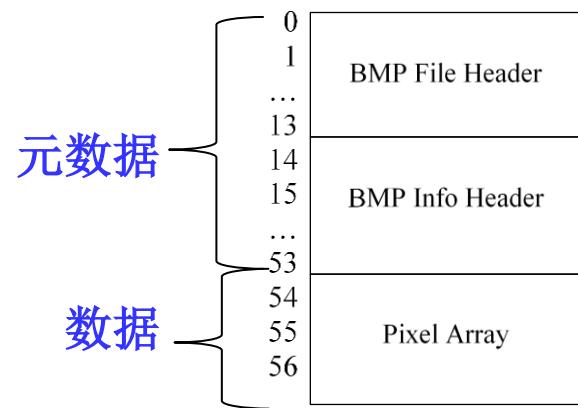
k 是4，因为每个字符是8比特，存在4个颜色中

- 计算机科学有数十种基本的定位机制
 - 本课程同学们只需理解一个，了解一个（URL），知道一个（指针）
- **操作：** 使用按位逻辑运算和mask(掩码)

信息隐藏实验的三层收获

- **基本要求：** 隐藏 HungLouMeng.txt 到 Autumn.bmp

- hide文本到图像；从图像show文本



- **掌握原理：** 定位到合适位置并操作该位置数据（理解这个CS的基本原理）

- 理解文本文件和图像文件的存储格式，即哪些字节（byte）存储数据、元数据
 - .txt文件每个字节是一个ASCII字符；.bmp文件采用bitmap格式，Pixel Array不从0地址开始
 - 将两个文件读入到两个变量，变量的数据类型都是字节切片（byte slice）

且能举一反三

- 能隐藏中文的红楼梦.txt吗？
 - 能隐藏到另一种图片格式吗（如Autumn.png）？音频文件呢？
 - 能将一张图片隐藏到另一张图片中吗？
- “.txt文件每个字节是一个ASCII字符”是狭义理解，可以是Unicode/UTF-8字符
- 关键是理解“两个变量的数据类型是字节切片（byte slice）”

- 受此实验启发产出新思想



U+1F339



U+2763

徐志伟老师:

您好!

我是课后提问您问题的那个同学。我的具体问题是:我们的英文文本可以被隐藏和复原,这较容易被理解。**但为什么我后来隐藏中文文本也可以被隐藏和复原?它的编码究竟是如何编制的?**

带着这个问题,我尝试探索其背后原理。我将“我爱计科导”文本实现了隐藏和复原。之后,首先,我在 show-2.go 的代码中添加下面的代码:

```
//print the first 15 bytes of the text
```

- 奇妙的思想火花, 值得研究
- 可能涉及职业操守和科学伦理方面的讨论, 参见中文教科书 1.3.3
“为什么未来不需要我们”

XXXX同学,

你提出了一个很妙的想法。已经有一些初步进展了, 同行称为“DNA存储”。。
如你有兴趣, 请搜索“DNA存储”, 或读一下《国家科学评论》的最新综述:

DNA Storage: Research Landscape and Future Prospects

<https://doi.org/10.1093/nsr/nwaa007>

徐志伟

-----原始邮件-----

发件人:<XXXXXX20@mails.ucas.ac.cn>

发送时间:2021-06-07 19:38:57 (星期一)

收件人: "徐志伟" <zxu@ict.ac.cn>

抄送:

主题: 回复: 关于计算机科学导论课后问题的自主探索与解答 (续)

(接上封邮件) 老师好, 在您的启发下我突然又想到一个问题, 就是生物的DNA是一串“嘌呤-嘧啶对”, 每个“嘌呤-嘧啶对”有四种不同的取法, 如果分别标记成0, 1, 2, 3的话, 那DNA就可以等效地用一串四进制数表示了, 这样的话, 也可以把这个大数隐藏到文件里面, 这是不是**对生物信息的隐藏**? ... 更进一步地, 我让这个DNA控制发育成一个生命体, 比如小白鼠, 看起来它跟其它的小白鼠没什么不一样, 但它却成了一个**“活体U盘”**, 无时无刻不携带着一段有用的信息! 甚至可以通过克隆这只小白鼠实现信息的克隆与传递。

也就是说, 我既可以**把一只小白鼠隐藏到《蒙娜丽莎》里面, 也可以把《蒙娜丽莎》隐藏到小白鼠里面?**

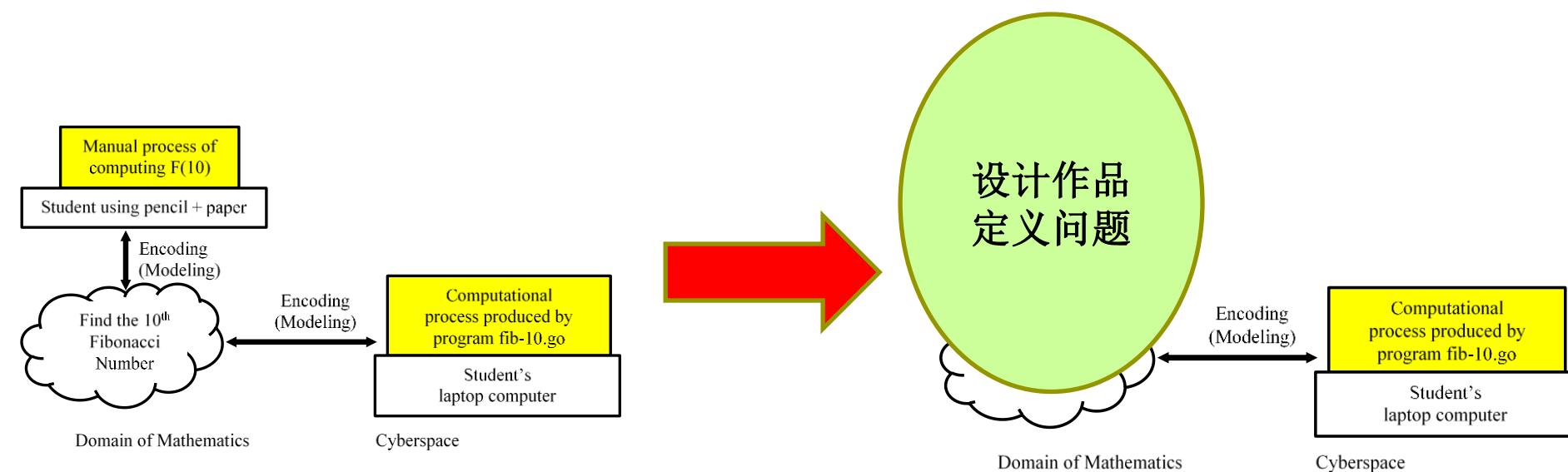
我觉得是一个奇妙的想法, 但不知道有啥用, 以及这玩意儿应该也很费钱.....跟您分享一下我的不成熟的想法:-)

XXXX

2021.6.7

什么是个人作品 (Personal Artifact)

- 创造性表达 (creative expression)，动态网页只是表现手段
- 充分利用个人的积累，充分发挥个人创造力
- 通过教学平台在全班分享，不只是学生个人和教学团队之间
 - 更加注意责任心，遵循职业操守 (ACM Code of Conduct)
 - 不能使用过多来自别人的素材
 - 必须有引用



什么是网页，什么是动态网页

- HTML/CSS/JavaScript入门知识，提升学习能力
- 学习方法建议
 - 将已学到的Go编程知识拓展到Web编程
 - 注意语法不同点，例如每条语句后要加“;”
 - 通过例子学习新知识；实验课有详解，与助教一起做一遍
 - 提升自己的学习能力与创造性表达
 - 可参考往届同学个人作品库
https://www.solid.things.ac.cn:7245/web_exp



Graphics credit:
Siyue Li
**50%时间创造
50%时间开发**

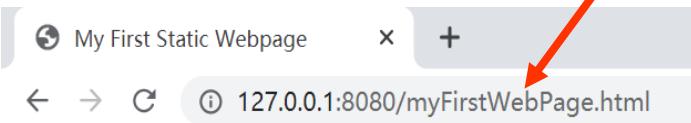
https://teacher.solid.things.ac.cn:7243/public/web/Kitty_Band.html

什么是Web客户端（客户机），什么是Web服务器？

- 客户机上运行**浏览器**，使用**网址**从Web服务器获取**网页文件**到客户机，再由浏览器处理并展示**网页**。（处理并展示=render）
 - 此时，Web服务器已经在运行

什么是Web客户端（客户机），什么是Web服务器？

- 客户机上运行**浏览器**，使用**网址**从Web服务器获取**网页文件**到客户机，再由浏览器处理并展示**网页**。（处理并展示=render）
 - 此时，Web服务器已经在运行
- 客户机和服务器可以是同一台计算机（如同学的笔记本电脑）
 - 此时Web服务器的IP地址是127.0.0.1，域名是localhost



Hello, World!

The HTML5 logo is shown on the left

假设客户机和服务器
都是某同学的笔记本电脑

```
> cat myFirstWebPage.html
<html>
  <head>
    <meta charset="utf-8">
    <title>My First Static Webpage</title>
  </head>
  <body>
    <h1> Hello, World! </h1>
    <p>
      
      The HTML5 logo is shown on the left
    </p>
    <script>
    </script>
  </body>
</html>
>
```

开发你的第一个网页（静态网页）

- 在你电脑的Linux环境中执行WebServer.go

```
> cat WebServer.go
package main
import "net/http"
func main() {
    http.HandleFunc("/", func(w http.ResponseWriter, r *http.Request) {
        http.ServeFile(w, r, r.URL.Path[1:])
    })
    http.ListenAndServe(":8080", nil)
}
```

```
> go build WebServer.go
```

```
> ./WebServer &
```

```
> [1] 442
```

```
>
```

```
>
```

```
> kill 442
```

Run program WebServer
in background

后台运行WebServer程序

WebServer 就绪
(442 is WebServer
process ID)

进程号、进程ID

Stop WebServer
终止ID为442的进程

你的第一个网页（html文档）及其网页显示

- 在浏览器网址栏输入下列URL（网址）

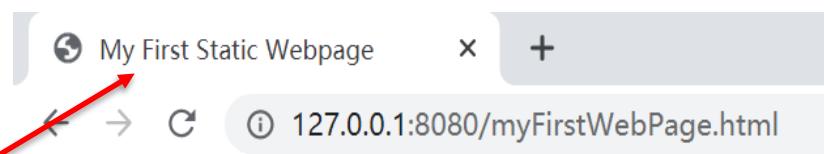
<http://127.0.0.1:8080/myFirstWebPage.html>

- 127.0.0.1 is localhost

```
> cat myFirstWebPage.html
<html>
  <head>
    <meta charset="utf-8">
    <title>My First Static Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1> 标题
    <p>段落
      
      The HTML5 logo is shown on the left
    </p>
    <script>
    </script>
  </body>
</html>
>
```

网页头

网页体



Hello, World!



The HTML5 logo is shown on the left

另一个
URL

HTML

被<html> ... </html>括起来
举一反三

Write your first Web program

- Enter “`http://127.0.0.1:8080/myFirstWebPage.html`” in a Web browser

```
> cat myFirstWebPage.html
<html>
  <head>
    <meta charset="utf-8">
    <title>My First Static Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>
      
      The HTML5 logo is shown on the left
    </p>
    <script>
    </script>
  </body>
</html>
>
```

Hello, World!



The HTML5 logo is shown on the left

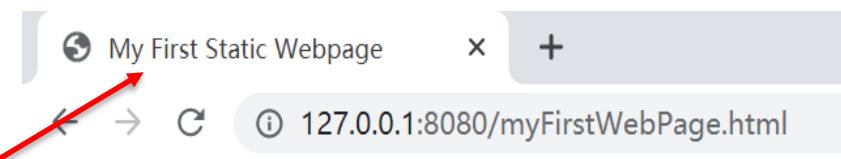
CSS 展示风格

图像浮在文字左边，30像素宽，30像素高

Write your first Web program

- Enter “`http://127.0.0.1:8080/myFirstWebPage.html`” in a Web browser

```
> cat myFirstWebPage.html
<html>
  <head>
    <meta charset="utf-8">
    <title>My First Static Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>
      
      The HTML5 logo is shown on the left
    </p>
    <script>
    </script>
  </body>
</html>
>
```



Hello, World!



The HTML5 logo is shown on the left

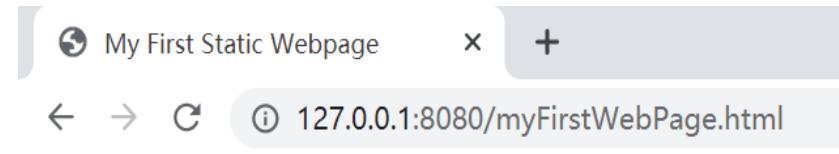
静态网页可不用JavaScript

No JavaScript code in this webpage
This is a static webpage

Write your first Web program

- Enter “`http://127.0.0.1:8080/myFirstWebPage.html`” in a Web browser
 - 127.0.0.1 is localhost

```
> cat myFirstWebPage.html
<html>
  <head>
    <meta charset="utf-8">
    <title>My First Static Webpage</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>
      
      The HTML5 logo is shown on the left
    </p>
    <script>
    </script>
  </body>
</html>
>
```



Hello, World!



The HTML5 logo is shown on the left

Web编程要点

- 在head里列出所有元素共性的东西
- 在body里列出网页(顺序)展示的元素
此例子共有三个元素：p元素包含img元素
- CSS指明元素的显示风格
- JavaScript代码操作元素（动态）

staticChildrensDay.html: code and output

- 使用JavaScript的例子

- Render the **Content** "2021.06.01";
- according to **Style** ~~-60px, purple~~
- at the place indicated by **Element ID** "childrensDay"

```
> cat staticChildrensDay.html
```

```
<html>
  <head>
    <meta charset="utf-8">
    <title>Display the date of next Children's Day</title>
  </head>
  <body>
    <h1 style="text-align: center">Date of Next Children's Day</h1>
    <p style="text-align: center" id="childrensDay" ></p>
    <script>
      var x = document.getElementById("childrensDay");
      x.style.fontSize = "60px";
      x.style.color = "purple";
      x.innerHTML = "2021.06.01";
    </script>
  </body>
</html>
>
```

Element ID 元素ID

Style 风格

Content 内容

Date of Next Children's Day

2021.06.01

- 定义段落元素的ID = childrensDay
- 使用document.getElementById函数，使得变量x指称该段落元素
- x.style指称该段落元素的风格
- x.innerHTML指称该段落元素的内容

staticChildrensDay.html: code and output

- 加上两条风格说明

```
> cat staticChildrensDay.html
```

```
<html>
  <head>
    <meta charset="utf-8">
    <title>Display the date of next Children's Day</title>
  </head>
  <body>
    <h1 style="text-align: center">Date of Next Children's Day</h1>
    <p style="text-align: center" id="childrensDay" ></p>
    <script>
      var x = document.getElementById("childrensDay");
      x.style.fontSize = "60px";
      x.style.color = "purple";
      x.innerHTML = "2021.06.01";
    </script>
  </body>
</html>
```

```
> Display Content
  with 60 pixels font size
  and purple color
```

- Render the **Content** "2021.06.01";
- according to **Style** 60px, purple
- at the place indicated by **Element ID** "childrensDay"

Element ID

Style

Content

Date of Next Children's Day

2021.06.01

ChildrensDay.html: code and output

- 使用JavaScript 产生动态网页
 - 显示的日期随时间正确变化
 - 本网页有多少元素? 2个。第二个元素是paragraph <p...></p>

```
> cat staticChildrensDay.html
<html>
```

```
...
<body>
  <h1 style="text-align: center">Date of Next Children's Day</h1>
  <p style="text-align: center" id="childrensDay" ></p>
  <script>
    var x = document.getElementById("childrensDay");
    x.style.fontSize = "60px";
    x.style.color = "purple";
    var date = new Date();
    var year = date.getFullYear();
    var month = date.getMonth() + 1;
    if (month >= 6) year = year + 1;
    x.innerHTML = "June 1, " + year;
  </script>
</body>
</html>
>
```

Element ID
Style
Content

Date of Next Children's Day

June 1, 2021

JavaScript is an object-oriented language

JS是面向对象的语言

- 对象：将数据结构和操作数据的方法放在一个抽象中
 - Object: Data structure + methods to operate the data structure
 - 使用点记号调用某一对象的某一方法
 - Use the dot notation to access methods of an object
 - date.getMonth: call the getMonth method of the **date** object
 - document.getElementById: call the getElementById method of the **document** object

```
<html>
<body>
<p id="myDate"></p>
<p id="myYear"></p>
<p id="myMonth"></p>
<script>
var date = new Date;
var year = date.getFullYear();
var month = date.getMonth() + 1;
document.getElementById("myDate").innerHTML = date;
document.getElementById("myYear").innerHTML = year;
document.getElementById("myMonth").innerHTML = month;
</script>
</body>
</html>
```

var date = **new** Date;

产生新对象并赋值（当前时刻）

This Web code displays three paragraphs:

- Full information of the current date
- Current year
- Current month

Fri Apr 16 2021 19:19:50 GMT+0800 (中国标准时间)
2021
4

Notes:

- Date is a system provided object
- document is a system provided object
- var date = new Date; create a new object
- Read textbook to see why month needs to add 1

系统提供很多
built-in对象，如
Date
Document

只需了解极少几个

The Web code can be rewritten as follows

- Note that the line of code
 - `document.getElementById("myDate").innerHTML = date;`is broken down into two shorter lines of code
 - `var x = document.getElementById("myDate");`
 - `x.innerHTML = date;`

Original Code

```
<html>
<body>
<p id="myDate"></p>
<p id="myYear"></p>
<p id="myMonth"></p>
<script>
var date = new Date;
var year = date.getFullYear();
var month = date.getMonth() + 1;
document.getElementById("myDate").innerHTML = date;
document.getElementById("myYear").innerHTML = year;
document.getElementById("myMonth").innerHTML = month;
</script>
</body>
</html>
```

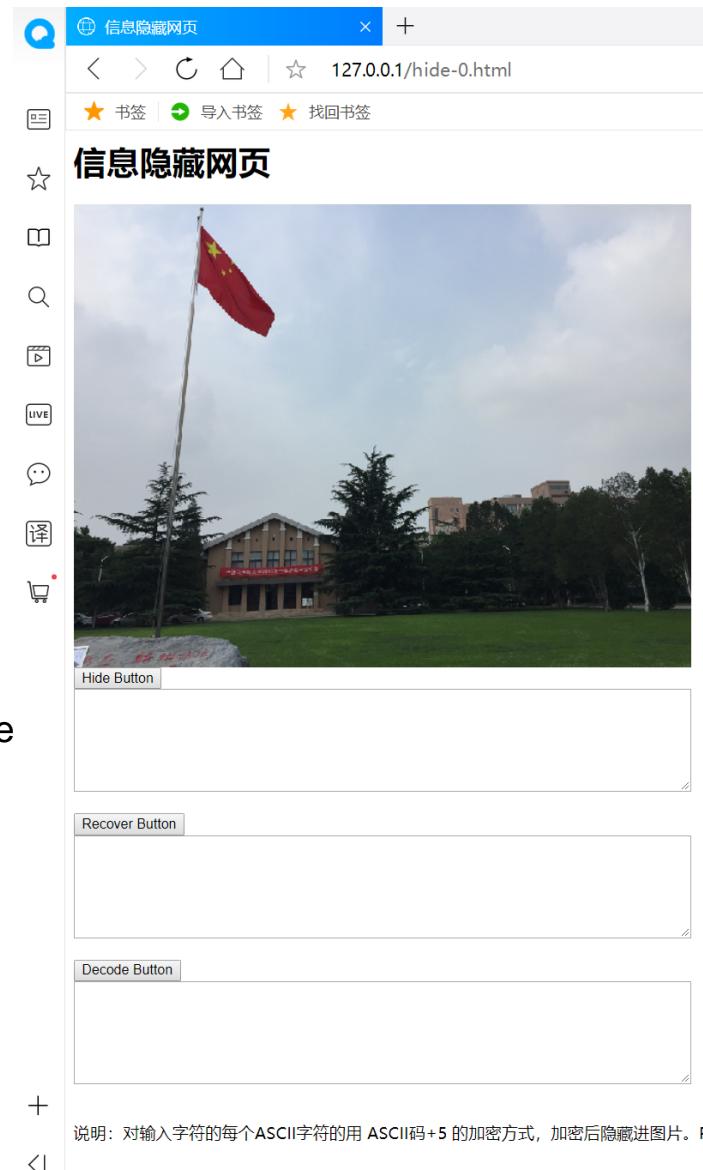
New Code

```
<html>
<body>
<p id="myDate"></p>
<p id="myYear"></p>
<p id="myMonth"></p>
<script>
var date = new Date;
var year = date.getFullYear();
var month = date.getMonth() + 1;
var x = document.getElementById("myDate");
x.innerHTML = date;
var y = document.getElementById("myYear");
y.innerHTML = year;
var z = document.getElementById("myMonth");
z.innerHTML = month;
</script>
</body>
</html>
```

Other HTML elements besides paragraph

● Canvas to draw a picture

```
<body>
.....
<canvas id="myCanvas" width="600" height="450" >
</canvas><br/>
<input type="button" value="Hide Button"
onclick="hide_info();"><br/>
<textarea id="hide_text" style="width: 600; height: 100px;">
</textarea><br/><br/>
.....
<script>
// load the image
var img = new Image();
var context1 = document.getElementById("myCanvas").ge
img.onload = function(){context1.drawImage(img,0,0);}
img.src = "./ucas.bmp";
.....
</script>
</body>
```



网页作品例子：信息隐藏

信息隐藏网页

信息隐藏网页

Hide Button

Recover Button

Decode Button

说明：对输入字符的每个ASCII字符的用 ASCII码+5 的加密方式，加密后隐藏进图片。Recover Button显示从图片里提取出的加密后的文本；Decode Button显示从图片里提取出的解密后的文本。

初始界面

网页作品例子：信息隐藏

信息隐藏网页

127.0.0.1/hide-0.html

书签 导入书签 找回书签

信息隐藏网页

Hide Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving.
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace.

Recover Button

Decode Button

说明：对输入字符的每个ASCII字符的用 ASCII码+5 的加密方式，加密后隐藏进图片。Recover Button显示从图片里提取出的加密后的文本；Decode Button显示从图片里提取出的解密后的文本。

网页作品例子：信息隐藏

信息隐藏网页

127.0.0.1/hide-0.html

书签 导入书签 找回书签

信息隐藏网页

Hide Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving.
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace.

Recover Button

tzw%knjisi%nx%~tzw%sjix%fsx|jwji3□Mj%nx%~tzw%knjqi%|nmhm%~tz%xt|%|nym&qt{j%fsi%wjfu%|n
ymymfsp1n{ns17%Fsi%nx%~tzw%gtwl%fsi%~tzw%knwjxni,j3□Ktw%~tz%htrj%yt%mn%|nym%~tzw%
zsljw1%fsi%~tz%jjp%mn%ktw%ujfbj7□

Decode Button

说明：对输入字符的每个ASCII字符的用 ASCII码+5 的加密方式，加密后隐藏进图片。Recover Button显示从图片里提取出的加密后的文本；Decode Button显示从图片里提取出的解密后的文本。

网页作品例子：信息隐藏

< > ⌂ ⌄ | ☆ 127.0.0.1/hide-0.html
★ 书签 | ⌂ 导入书签 ★ 找回书签

信息隐藏网页



Hide Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving.
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace.

Recover Button

```
^tzw&knwjsi%nx%~tzw&ejjix%fsx|jwji3□Mj%nx%~tzw&knjqi%|mnhm&~tz&xt|%|nym&qat{jq%fsi%kwjfu%|n
ym&ymfspxln|ns17□Fs i%mj%nx%~tzw&tfwi%fsi%~tzw&knwjxn1j3□Ktw%~tz&htrj%yt%mr%|nym&~tzw&km
zs1jw1%fsi%~tz&xjjp%mr%ktw&ujfhj7%
```

Decode Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving2
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace2

点击Decode按钮
恢复明文

信息隐藏网页



Hide Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving.
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace.

Recover Button

`^tzw%knwjsi%nx%~tzw%sjjix%fsx|jwji3□Mj%nx%~tzw%knjqi%|mnhm%~tz%xt|%|nym%qt{J%fsi%wvjfu%|n
ym%ympspxlnfn17□Fsi%mj%nx%~tzw%gtfwi%fsi%~tzw%knwjxni3□Ktw%~tz%htrj%yt%mnur%|nym%~tzw%
zsljw1%fsi%~tz%xjjp%mnur%ktw%ujfhj7□`

Decode Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving2
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace2

```

<html>
  <head>
    <meta charset="UTF-8">
    <title>信息隐藏网页</title>
  </head>
  <body>
    <h1>信息隐藏网页</h1>
    <canvas id="myCanvas" width="600" height="450" ></canvas><br/>
    <input type="button" value="Hide Button" onclick="hide_info();"><br/>
    <textarea id="hide_text" style="width: 600px; height: 100px;"></textarea><br/><br/>
    <input type="button" value="Recover Button" onclick="show_info(0);"><br/>
    <textarea id="show_info_0" style="width: 600px; height: 100px;"></textarea><br/><br/>
    <input type="button" value="Decode Button" onclick="show_info(1);"><br/>
    <textarea id="show_info_1" style="width: 600px; height: 100px;"></textarea><br/><br/>
    <p>说明：对输入字符的每个ASCII字符的用 ASCII码+5 的加密方式，加密后隐藏进图片。Recover Button显示从图片里提取出的加密后的文本；Decode Button显示从图片里提取出的解密后的文本。</p>
    <script>
      // load the image
      var img = new Image();
      var context1 = document.getElementById("myCanvas").getContext("2d");
      img.onload = function(){context1.drawImage(img,0,0);}
      img.src="./ucas.bmp";
      .....
    </script>
  </body>
</html>

```

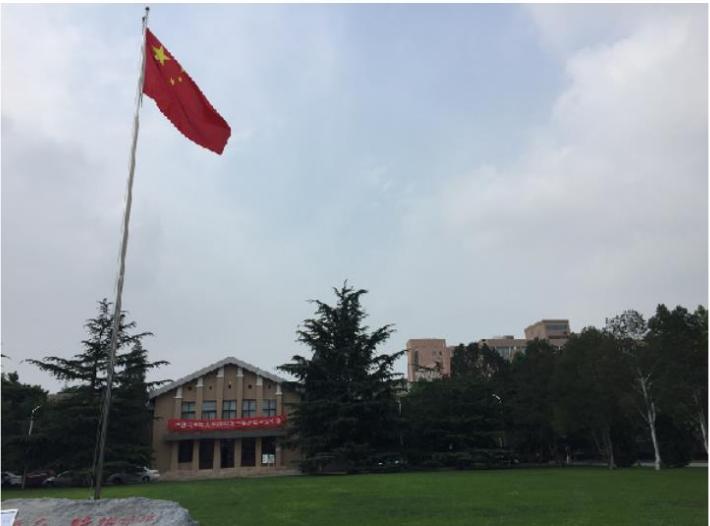
More details
on next page

Go Code

```
func modify(data int, array []byte, size int) {
    for i := 0; i < size; i++ {
        v := byte(data & 0x3)
        array[i] = array[i] & 0xFC
        array[i] = array[i] | v
        data = data >> 2
    }
}
```

Go Code

信息隐藏网页



Hide Button
Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving.
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace.

Recover Button

^tzw%knwjsi%nx%~tzw%sjjix%fsx|jwj13□Mj%nx%~tzw%knjqi%|mnhm%~tz%xt|%|nym%qt{j%fsi%wvjfu%|n
ym%ymfspxlfn17□Fsi%mj%nx%~tzw%gtfw%fsi%~tzw%knwjxni3□Ktw%~tz%htrj%yt%mn%|nym%~tzw%
zsljw1%fsi%~tz%xjjp%mn%ktw%ujfhj?□

Decode Button

Your friend is your needs answered.
He is your field which you sow with love and reap with thanksgiving2
And he is your board and your fireside.
For you come to him with your hunger, and you seek him for peace2

```
func main() {
    txt_path           := "./Richard_Karp.txt"
    src_img_path       := "./ucas.bmp"
    dest_img_path      := "./modified_ucas.bmp"
    d, _ := ioutil.ReadFile(src_img_path)          // read ucas.bmp into d
    t, _ := ioutil.ReadFile(txt_path)               // read Richard_Karp.txt into t
    modify(len(t), d[S : S+T], T)                  // hide length of Richard_Karp.txt
    for i:=0; i<len(t); i++{                      // hide contents of Richard_Karp.txt
        offset := S+T+(i*4)
        modify(int(t[i]),d[offset:offset+C],C)
    }
    ioutil.WriteFile(dest_img_path, d, 0666) // output to modified_ucas.bmp
}
```

```
function hide_info(){
    // 1. get original pixel_array
    var context1 = document.getElementById("myCanvas").getContext("2d");
    var img_data = context1.getImageData(0,0,600,450);
    var pixel_array = img_data.data;
    // 2. get input string
    var str = document.getElementById("hide_text").value;
    // 3. hide length
    modify(str.length, pixel_array, 16, 0);
    // 4. hide string
    for(var i=0;i<str.length; i++){
        var ch = str[i].charCodeAt() + 5;      // +5是加密操作
        modify(ch, pixel_array, 4, 16+4*i);
    }
    // 5. show modified image
    context1.putImageData(img_data,0,0);
}

function modify(data, array, size, left) {
    for (var i = left; i < left+size; i++) {
        var v = data & 0x3;                // retain last 2 bits of data
        array[i] = array[i] & 252;         // clear last 2 bits of array[i]
        array[i] = array[i] | v;          // set last 2 bits of array[i] with those of data
        data = data >> 2;                // repeat with the next 2 bits
    }
}
```

JavaScript Code

说明：对输入字符的每个ASCII字符的用 ASCII码+5 的加密方式，加密后隐藏进图片。Recover Button显示从图片里提取出的加密后的文本；Decode Button显示从图片里提取出的解密后的文本。