Trần Duy Ngọc C1220I1

60 câu hỏi trắc nhiệm Module 1  
  
Num1 :  
Num2 : Thuật toán/giải thuật là các chỉ thị để giải quyết vấn đề   
  
Num3 : Pesudo – code và Flowchart : Pesudo - code dễ trình bày theo văn nói nhưng khi nhìn lại dài dòng và khó nhớ Flowchart sử dụng theo dạng sơ đồ rẽ nhánh gọn gàng bao quát và rất dễ hiểu nếu biết các quy tắc các (khối , đường đi) minh họa ngược lại không biết không thể hiểu được  
  
Num4:Trình bài giải thuật trước khi code để xác định các bước quan trọng cần thực hiện,bước nào trước bước nào sau,chia bài toán ra nhiều vấn đề nhỏ hơn để giải quyết trình tự không bị sót yêu cầu bài  
  
Num5: Flowchart có tính hệ thống dễ nhìn tổng quan với bài toán dài không bị rối dễ hiểu vì nếu sử dụng Pesudo-code mỗi người 1 lối mô tả khác nhau   
  
Num6: Giải thuật cần hữu hạn bước để khi code được tối ưu nhất ngắn nhất mà hiệu quả nhất , tiết kiệm thời gian và tránh gặp sai xuất hiện ở những đoạn không tối ưu ảnh hưởng xấu  
  
Num7:   
  
 : Lệnh nhập xuất : Đường đi của lưu đồ   
  
  
 : Lệnh bắt đầu , kết thúc   
   
  : Nối hai phần của trương trình  
 : Các bước tính toán  
  
  
 : Quyết định và rẽ nhánh  
  
  
  
  
  
  
  
Num8: Giải thuật giải PT bậc nhất

a, b, c

aX+b=c

a = 0

c-b=0

0 Nghiệm

Vô số Ngiệm

1 nghiệm  
x= (c-b)/a

Num9: VCS (phần mềm)là 1 bên thứ 2-3 giúp chúng ta quản lý những thay đổi về source code của dự án theo thời gian giúp dê dàng quản lý thay đổi lưu trữ   
  
Num10: VCS Tập trung gồm 1 main server mọi hoạt động đều thông qua nó   
 DVCS lại không bắt buộc mọi thứ phải đưa về 1 main server , mỗi thành viên có muột nơi lưu trữ riêng và dễ dàng chia sẻ , update source-code  
  
Num11: Git là một phần mềm DVCS .Dùng git cần thiết vì : với git chúng ta dễ dàng lưu trữ thay đổi chia sẻ source-code, hỗ trợ làm việc nhóm Git có cảnh báo khi source-code bị đè lên nhau giúp tránh lỗi khi nhiều người làm cùng 1 công việc

Num12: Git là DVCS quản lý mã nguần phân tán

Num13:Repository là nơi lưu trữ source – code và với git Local Repo là nơi lưu những source-code của máy tính cá nhân . khi đưa Local Repo lên trên “git” thì Local Repo này chuyển thành Remote Repo

Num14: để cập nhật những thay đổi từ cả local và remode  
  
Num15: Push của “git”là cập nhập thay đổi của local repo lên remode repo

Num16: Pull là ngược lại của Push :cập nhật thay đổi từ remode repo về local repo

Num17: Clone là sao chép 1 remode repo trên “git” về máy . cú pháp : git clone + Link git cần sao chép

Num18: “ git init “ khởi tạo 1 repository 1 project mới hoặc đã có  
 : “ git clone” câu 17   
 : “git add “   
 : “git push” câu 15  
 : “git pill “ câu 16  
 : “git commit” là cách git chụp lại những thay đổi trong thư mục làm việc tại thời điểm “git commit”  
 : “git remote “ kiên kết với địa chỉ trên “git” để lúc “git push” hoặc “git pull” hoặc “git clone” sẽ hướng vào địa chỉ đó   
 : “git remove” thu hồi lại liên kết với địa chỉ trên “git”  
 : “git status” để kiểm tra trạng thái file đã thay đổi trong thư mực vào những lần commit”

Num19: WWW là viết tắt của World Wide Web thế giới rộng lớn của các trang web quy mô toàn cầu

Num 20: HTML là ngôn ngữ lập trình nhằm mục đích tạo ra các trang web trên WWW

Num21: Các thẻ làm việc với văn bản là

Thẻ <h>Tiêu đề</h> , <h1>tiêu đề</h1> , <h2>Tiêu đề</h2> , <h3>Tiêu đề</h3>….<h6>Tiêu đề</h6>

Thẻ <p>Văn bản</p>

Thẻ <strong>phần nội dung này sẽ được in đậm</strong>

Thẻ <u>phần nội dung này sẽ được gạch chân</u>

Thẻ <i>phần nội dung này được in nghiêng</i>

Thẻ <strike>phần nội dung này sẽ được gạch ngang</strike>

Thẻ <em>nội dung được nhấn mạnh</em>

Thẻ <br> để ngắt dòng

Thẻ <mark>chữ viết ở đây được tô sáng</mark>

Thẻ <quote> thể hiện câu nói dạng quote,trích dẫn</quote>

Thẻ <cite>tên tác giả của trích dẫn</cite>

Num22: Thẻ Div thẻ Span đều để chứa nội dung nhưng thẻ “div” là loại block và “span” là loại inline  
khác nhau là khi sử dụng sau Thẻ “div” nội dung sẽ bị đẩy xuống dòng dưới không cho sang bên cạnh và “span” sẽ cho phép làm điều đó

Num23: URL viết tắt của Uniform Resource Locator là địa chỉ của 1 tài nguyên trên duy nhất trên web   
 URL tuyệt đối là đường dẫn chỉ ra vị trí chính xác của file,thư mục  
 URL tương đối chỉ dẫn được đến vị trí cần đến khi ở trong cùng một thư mục,nếu URL tương   
 dùng vào thư mực khác sẽ ko ra kết quả  
  
Num24: Siêu liên kết : Hyperlink – ở bất cứ đâu trỏ vào a =” hình ảnh hay đường link ,nút …”   
 con trỏ chuột biết thành hình bàn tay,ấn vào và chạy ra 1 trang web khác thì a là “hyperlink”  
  
  
Num25: noteeeeeee  
  
  
  
  
Num26: Để liên kết đến 1 phần trên trang khác sử dụng thẻ <a> nút ấn để đến trang đã liên kết</a>  
 <a href=”đường link muốn liên kết” > nút ấn để đến trang đã liên kết </a>  
  
Num27: Internal link là đường link liên kết đến một phần trên cùng 1 trang web cùng 1 tên miền còn   
 External link là đường link liên kết đến một trang web khác,khác tên miền

Num28: Có nhiều loại danh sách tùy vào mục đích tạo ra chính thì là 2 loại danh sách có thứ tự<ol>  
 và danh sách không theo thứ tự <ul>  
  
Num29: <ol type=”A” start=”3”> </ol>

Num30: Form để người dùng điền thông tin vào và ta có thể lấy thông tin được điền đó về server

Num31: Thuộc tính action có giá trị là 1 URL là nơi thông tin người dùng điền vào form sẽ được gửi đến

Num32: Phương thức GET gửi dữ liệu tường mình( hiện ngay trên URL)  
 webbrower sẽ cached lại nên gửi dữ liệu sẽ nhanh nhưng không bảo mật  
  
 Phương thức Post do server luôn thực thi và trả kết quả cho client – quá trình lâu hơn  
 bảo mật hơn   
 Gửi dữ liệu thay đổi liên tục,cần bảo mật xài Post,xài GET không an toàn.  
 Ngược lại dữ liệu ít thay đổi vẫn xài Post được truy xuất và quản lý sẽ không nhanh bằng GET  
  
  
Num33: nút bấm type là “submit” : thực hiện gửi thông tin về URL dán ở action  
 “reset” : reset lại toàn bộ thông tin người dùng nhập vào form về mặc định  
 “button” : thực hiện khối lệnh tùy vào mục đích muốn nút đó làm gì (JSC )  
  
Num34: Để submit form tới trang CodeGym.vn thì ta để action=”CodeGym.vn”  
 : <form action=”CodeGym.vn” method=”post”>  
  
Num35: <td colspan=”3”> </td>  
  
Num36:<td rowspan=”2”> </td>  
  
Num37: cellpadding để tạo khoảng cách giữa nội dung trong ô so với đường viền ,  
 cellspacing để tạo khoảng cách giữa viền của 1 ô và viền của bảng  
  
Num38: Có 2 cách nhúng Javascript vào HTML   
 Cách 1: mở thẻ <script></script> ngay trong HTML (ngoài body) và thêm phần JSC  
 Cách 2: mở thẻ <script></script> ngay trong HTML trỏ đến file chứa JCS  
  
Num39: Phát biểu: “JavaScript là một ngôn ngữ script dựa trên đối tượng nhằm phát triển các ứng dụng Internet dựa trên client và server” là sai vì JSC giúp phát triển các ứng dụng Internet “chạy trên” client và sever  
  
  
Num40: Đúng  
  
Num41: CamelCase và camelCase là quy tắc đặt tên trong lập trình (con lạc đà)  
 CamelCase : viết hoa các chữ cái đầu của từ khóa  
 camelCase : chữ cái đầu của từ khóa đầu không viết hoa chữ cái đầu của từ khóa tiếp theo   
 sẽ viết hoa  
  
Num42: Biến là tên gọi ta gán cho giá trị để dễ sử dụng ,dễ gọi,dễ nhớ , ngắn gọn code với 2 loại biến  
 (local)-biến cục bộ và (global)-biến toàn cục  
 Biến cục bộ : biến được khai báo bên trong 1 hàm. Ra ngoài hàm biến đó không có ý nghĩa  
 Biến toàn cục : biến được khai báo bên ngoài , không nằm trong hàm cụ thể nào cả  
  
  
Num43: Var là phương thức khai báo biến toàn cục (global)   
 let là phương thức khai báo biến cục bộ (local)  
 const là phương thức khai báo biến không đổi   
  
Num44: Phát biểu “JavaScript là ngôn ngữ định kiểu yếu hay động”  
 Tức khi khai báo biến không cần khai báo kiểu dữ liệu của biến , kiểu dữ liệu sẽ phụ thuộc vào giá trị  
 ta gán cho biến  
Num45: Các kiểu dữ liệu nguyên thủy:   
 +Float : sô thực 32bit  
 +Chart: ký tự đơn dạng 16bit  
 +Int : số nguyên   
 +Double : số thực (lớn hơn float) 64bit  
 +Boolean: 1 trong 2 giá trị True hoặc False

+String : chuỗi   
 +null : giá trị rỗng hoặc không tồn tại   
 +undefind : kiểu dữ liệu không xác định (=””)  
  
Num46: Typeof dùng để xác định kiểu dữ liệu của biến   
 “let x=10” thì typeof của x là kiểu number  
  
Num47: Có 3 cách hiển thị thông báo trong JSC  
 + alert(“nội dung”) chỉ hiển thị thông báo  
 +confirm(“nội dung “) hiển thị nội dung thông báo và hai nút “ok” và “cancel” rẽ nhánh if else  
 +prompt(“nội dung yêu cầu”) hiển thị thông báo yêu cầu người dùng nhập dữ liệu vào   
  
Num48:Các loại toán tử trong JSC  
 Toán tử typeof : trả về kiểu dữ liệu cần kiểm tra của một biến  
 Toán tử gán :

|  |  |  |
| --- | --- | --- |
| Toán tử | Ví dụ | Ý nghĩa |
| = | Haha=1 | Gán 1 vào Haha |
| += | x+=y | x = x+y |
| -= | x-=y | x=x-y |
| \*= | x\*=y | x=x\*y |
| /= | x/=y | x=x/y |
| %= | X%=y | X=x%y |

Toán tử số học:

|  |  |
| --- | --- |
| Toán tử | Mô tả |
| + | Cộng |
| - | Trừ |
| \* | Nhân |
| / | Chia |
| % | Chia lấy phần dư |
| ++ | Công thêm 1 |
| -- | Trừ đi 1 |

Toán tử so sánh :

|  |  |  |
| --- | --- | --- |
| Toán tử | Diễn tả | Ví dụ |
| == | So sánh bằng | 5==10 false |
| === | Giống nhau (cùng kiểu dữ liệu và giá trị) | null===undefind false |
| != | Khác giá trị | 5!=10 true |
| !== | Khác giá trị và kiểu | 5!==10 false |
| > | Lớn hơn | 10>5 true |
| < | Bé hơn | 10<5 false |
| >= | Lớn hơn hoặc bằng | 5>=10 false |
| <= | Bé hơn hoặc bằng | 5<=10 true |

Toán tử logic:

|  |  |
| --- | --- |
| Toán tử | Diễn tả |
| && | Phép “và” trả về True khi a true && b true |
| || | Phép “hoặc” trả về True khi a true || b true |
| ! (a=5) | Phủ định trả về giá trị ngược lại : !a # 5 |

Num49: “==” là toán tử bằng so sánh giá trị Vd: math.cell(1,3)==1 true  
 “===”là toán tử giống nhau (cả về giá trị và kiểu dữ liệu) VD:math.cell(1,3)===1 false  
  
Num50: && trả là true khi cả 2 đều phải đúng nếu 1 trong 2 sai thì kết quả sẽ trả ra false  
 || trả ra true khi 1 hoặc cả 2 đều đúng nếu cả 2 sai thì trả ra false

Num51: +0===-0 true  
  
Num52: “++” tăng thêm 1 “--" trừ đi 1 “-“ là toán tử trừ  
   
Num53: ++a là cộng luôn 1 vào a còn a++ là khi kết thúc dòng lên sẽ + 1 cho a ( số tăng dần)  
  
Num54: Câu lệnh điều kiện rẽ nhánh là câu lệnh có dạng nếu … thì   
  
Num55: Có những loại là : if-else ; if lồng (bậc thang) và switch-case  
  
Num56: Câu lệnh if

Đúng Sai  
  
  
  
  
  
  
  
  
  
  
Num57: switch-case chính là “if lồng if” giúp đỡ rối ngắn gọn   
   
Num58: else là 1 tùy chọn (optional) của câu lênh if vì:trong câu lệnh if không tùy trường hợp sẽ không   
 cần else  
   
Num59: Sử dụng switch để thay if lồng ( có nhiều điều kiện )  
  
Num60: Câu lệnh break để kết thúc 1 điều kiện trong switch ( 1 case )   
 Nếu không có “break” giá trị sẽ không trả cho điều kiên đó ( case đó ) mà sẽ nhận giá trị của   
 case cuối cùng

Điều kiện

Câu lệnh tiếp

Câu lệnh