|  |  |  |  |
| --- | --- | --- | --- |
| Thành viên/Tên nhóm | Nhóm 1 | Nhóm 2 | Nhóm 3 |
| 1 | Cường | Hải | Thảo |
| 2 | Luy | Tuấn | Linh |
| 3 | Việt | Dương | Chơn |

|  |  |
| --- | --- |
| **Mô tả thuật toán bằng Pseudo code & Flowchart** | |
| Lập trình là gì | là quá trình tạo ra tập các chỉ dẫn (instruction) để ra lệnh cho máy tính hoàn thành một công việc (task) nào đó |
| Ngôn ngữ lập trình là gì | là phương tiện để lập trình viên viết ra các chỉ dẫn cho máy tính |
| Thuật toán là gì | Là nhiều chỉ thị nhằm giải quyết một vấn đề |
| Các cách biểu diễn giải thuật thông dụng | * • Mã giả (Pseudo-code) * • Lưu đồ (Flowchart) * • Ngôn ngữ lập trình |
| Mã giả là gì ? | Là thuật toán được biểu diễn bằng ngôn ngữ tự nhiên được Liệt kê tuần tự các bước |
| Lưu đồ là gì ? | Là thuật toán được biểu diễn bằng sơ đồ hình khối, mỗi khối qui định một hành động. |
| Hình bầu dục | Chứa các lệnh để Begin hay End 1 chương trình |
| Hình chữ nhật | Chứa các lệnh để thực hiện các phép toán |
| Hình bình hành | Chứa các lệnh để nhập xuất dữ liệu |
| Hình thoi | Chứa các lệnh để quyết định chương trình hoặc rẽ nhánh |
| Đốt | Lệnh nối 2 tháo tác trong chương trình |
| Mũi tên | Trình tự thực hiện các thao tác theo hướng mũi tên |
| Cấu trúc lựa chọn | Là 1 lệnh Lựa chọn một công việc để thực hiện căn cứ vào một điều kiện nào đó. |
| Cấu trúc lựa chọn lồng nhau | Là 2 hoạc nhiều hơn  cấu trúc lựa chọn được lồng vào nhau: IF   IF  Else Else |
| Cấu trúc lặp | Là 1 lệnh Thực hiện lặp lại một công việc không hoặc nhiều lần căn cứ vào một điều kiện nào đó. |
| AND / OR | Là lệnh toán tử. Thêm diều kiện cho cấu trúc lựa chọn |
| Giải thích tính tổng hai số | * let a=5,b=6; * console.lg(a+b) |
| Giải thích Kiểm tra số chẵn | if(a%number==0)chan |
| Giải thích Hiển thị 1000 từ Scooby |  |
| EX: Bài tập Nhân a,b | - |
| EX: Nếu 18 tuổi uống bia, ngược lại uống rượu |  |
| EX: In ra 10 lần câu xin chào |  |
| EX: Tỉnh tổng của 1 dãy số |  |
| **Git & HTML :** | |
| Quản lý mã nguồn là gì |  |
| Git là gì ? | la phan mem quan ly ma nguon |
| GitHub là gì ? | ma nguon luu tru |
| Snapshot là gì ? | giup minh gui code len remote |
| Commit là gì ? | comment |
| Repository là gì ? | kho luu tru |
| Clone là gì | sao chep ma nguon |
| Pull là gì ? | dong bo remote ve local |
| Push là gì ? | dong bo local len remote |
| Các lệnh git cơ bản | git init  git commit -m  git branch  git push  git  pull |
| HTML là gì ? | tao ra cac thanh phan cua trang web |
| Các thẻ HTML thông dụng |  |
| **HTML Form & Table :** | |
| action | quy định nơi dữ liệu gửi đến |
| Form dùng để làm gì ? | cho phép người dùng gửi dữ liệu lên sever |
| Các thuộc tính cơ bản của thẻ form | action va method(get post) |
| Thẻ input dùng để làm gì | tạo ra trường nhập dữ liệu |
| Các loại input | Thuộc tính name: tên của trường dữ liệu •  Thuộc tính disabled: vô hiệu hóa trường dữ liệu, không cho người dùng tương tác • Thuộc tính value: giá trị của trường dữ liệu •  Thuộc tính type: loại trường dữ liệu (text, password…) |
| Thẻ button dùng để làm gì | button được sử dụng để tạo một nút, nút này có thể được gắn với sự kiện của Javascript để xử lý các thao tác của người dùng |
| Thẻ textarea dùng để làm gì |  |
| Sử dụng thẻ select,option |  |
| Phương thức GET | du lieu duo gui di hien thi tren thanh dia chi cua trinh duyet |
| Phương thức POST | du lieu gui di khong han che va khong hien thi tren thanh dia chi trinh duyet |
| Table dùng để làm gì | taoj bang |
| Thẻ tr dùng để làm gì | tao hang  rowspan |
| Thẻ td dùng để làm gì | noi dung tung cot  colspan gop cot |
| Thẻ th dùng để làm gì | hien thi tieu def |
| **Tổng quan JavaScript :** | |
| Javascript là gì ? | la ngon ngu lap trinh cho phep cacs chuc nang phuc tap cua trang web  nn lap trinh viet ra ngon ngu chi dan may tinh |
| Cách Nhúng mã Javascript vào file HTML | c1 nhung theo kieu internal dat the script trong head hoac body bat cu cho nao ban muon  c2 nhung theo kieu inline viet ma script ngay khi khai bao html  <button onclick=”alert(“gsgdgas”)”>  c3 nhung theo kieu external nhung rieng biet |
| Quy tắc đặt tên file | khong su dung dau cach khi dat ten fife  ten file phan biet chu hoa thg  khoong su dung kis tu dac biet chi su dung gach duoi gavh noi vaf dau cham de su dung phan mo rong  giu ten file ban ngan |
| Cách bắt sự kiện từ thuộc tính | c1 bat su kien |
|  |  |
| Cách bắt sự kiện qua eventlistener | *element*.addEventListener(*event*, *function*) |
| Sự kiện onchange | Sự kiện thay đổi xảy ra khi giá trị của một phần tử đã được thay đổi. |
| Sự kiện onmouseover và onmouseout | Sự kiện onmouseout xảy ra khi con trỏ chuột được di chuyển ra khỏi một phần tử hoặc ra khỏi một trong các phần tử con của nó  Sự kiện onmouseover xảy ra khi con trỏ chuột được di chuyển vào một phần tử hoặc vào một trong các phần tử con của nó. |
| Sự kiện onclick |  |
| alert() | hien thi  hop thoai ra man hinh |
| document.write() |  |
| console.log() |  |
| Tìm phần tử dựa vào thuộc tính id="nut" | Thuộc tính id đặt hoặc trả về id của một phần tử  ID phải là duy nhất trong một trang và thường được sử dụng để trả về phần tử bằng phương thức [document.getElementById ()](https://www.w3schools.com/jsref/met_document_getelementbyid.asp) . |
| innerHTML | Thuộc tính innerHTML trả về:  Nội dung văn bản của phần tử, bao gồm tất cả các thẻ khoảng cách và thẻ HTML bên trong. |
| innerText | Thuộc innerTexttính đặt hoặc trả về nội dung văn bản của một phần tử |
| Hàm trong javascript |  |
| **Biến, kiểu dữ liệu và toán tử :** | |
| Biến là gì ? | la mot ten goi gan cho 1 du lieu |
| Hằng là gì ? | Hằng là "tên biểu tượng" đại diện cho một giá trị không thay đổi trong chương trình    const <tên hằng> = <giá trị của hằng>; |
| Cú pháp khai báo biến | <kiey du lieu> <ten bien>=<gia tri bien>; |
| Quy tắc đặt tên biến | Bắt đầu bằng ký tự hoặc kí tự gạch dưới \_ (không được bắt đầu bằng số).  Không chứa các kí tự đặc biệt như & \* ( , ).  Tên biến không được trùng với từ khóa như var let for if ...  this để đại diện cho một đối tượng (Object |
| Các kiểu dữ liệu của javascript | * chuoi, boolean,number,undefiner,array,object * object  một tập hợp gồm những cái tên và mỗi cái tên sẽ chứa đựng một giá trị dữ liệu.   cus phap obj    var SinhVien = {          name:"Nhân",          gender:"Nam",          year:1993      }  doi tuongg la sinh vien  thuoc tinh la name gender year  gia tri thuoc tinh nhan nam 993   * khi khai bao bien khong gan gia tri se tra ve undefined * - Mảng là một loại biến đặc biệt có thể lưu trữ nhiều giá trị đồng thời, mỗi giá trị được gọi là một phần tử mảng.      var mobile = ["HTC","Nokia","SamSung"]; |
| Khai báo được các kiểu dữ liệu |  |
| Các loại toán tử | thuc hien 1 hanh dong cu the |
| Toán tử so  học | +-\*/\*\*++-- |
| Toán tử gán | gán giá trị cho một biến  = +=-=/=%= |
| Toán tử cộng chuỗi | var a=”name”  var b=”fist”  console.log(a+b) |
| Toán tử so sánh | danh gia muc do tuong quan cua cac gia tri  == bang  ===  != khac  >=<= |
| Toán tử logic | dung trong cac bieu thuc logic(true/false)  && toan tu va  || toan tu hoac  ! toan tu phu dinh |
| Toán tử typeof | lay ve kieu du lieu cua mot bien  vd typeof ‘’hai”  return string |
| **Cấu trúc điều kiện :** | |
| Khi nào cần dùng cấu trúc điều kiện | sử dụng if để xác định một khối mã được thực thi, nếu một biểu thức điều kiện được đánh giá là true  Sử dụng else để xác định một khối mã được thực thi, nếu điều kiện được chỉ ra trong if là sai  Sử dụng else if nhằm xác định một điều kiện mới để kiểm tra, nếu các điều kiện trước đó là sai  Sử dụng switch để xác định nhiều khối mã luân phiên được thực thi |
| So sánh giữa if bậc thang và switch-case | if (biểu thức điều kiện 1) {  khối mã được thực thi nếu điều kiện 1 là đúng  } else if (biểu thức điều kiện 2) {  khối mã được thực thi nếu điều kiện 1 là sai và điều kiện 2 là đúng  } else {  khối mã được thực thi nếu điều kiện 1 và điều kiện 2 đều sai  }  Sử dụng lệnh switch để chọn một trong số nhiều khối mã được thực thi.  Sử dụng mở rộng else if để chỉ định một điều kiện mới nếu các điều kiện trước đó là sai.  do {      // some thing  }  while (condition); |
| Cú pháp của câu lệnh if | if (biểu thức điều kiện) {  khối mã được thực thi nếu điều kiện là đúng  } |
| Cú pháp của câu lệnh if – else | if (biểu thức điều kiện) {  khối mã được thực thi nếu điều kiện là đúng  } else {  khối mã được thực thi nếu điều kiện trên là sai  } |
| Cú pháp câu lệnh switch-case | switch(biểu thức) {    case giá trị 1:  // khối lệnh sẽ được thực thi  break;    case giá trị 2:  // khối lệnh sẽ được thực thi  break;    default:  // khối lệnh mặc định sẽ được thực thi nếu không chọn được trường hợp nào    }  while (condition){      // do something  } |
| Biểu thức điều kiện | - |
| Switch-case là gì ? | Lệnh switch được sử dụng để thực hiện các hành động khác nhau dựa trên các điều kiện khác nhau. |
| Khi nào sử dụng switch-case và if | swith-case: so sanh bằng hoặc khác nhau ìf- else: ss tất cả |
| ++A |  |
| A++ |  |
| **Vòng lặp** | |
| Vòng lặp là gì | Vòng lặp là một cấu trúc lập trình cho phép thực hiện một đoạn mã lặp đi lặp lại nhiều lần. |
| Ví dụ về vòng lặp | lay phaan hoa. đi thang 5 buoc rẽ trái |
| Cú pháp vòng lặp for | for( câu lenh dc thực thi 1 lần khi chạy vòng lặp **;** biểu thức điều kiện xác dinh xem vòng lặp có dc thuc thi hay không **;** các câu lệnh duoc thực thi sau mỗi lần lặp){ |
| *Ex: Hiển thị bảng cửu chương của 5* | for(i=2;i<=9;i++){           for(j=1;j<=9;j++){      document.wire(i +“\*”+j+”=”+(i\*j));      }  } |
| Cú pháp vòng lặp while |  |
| *Ex: Nhập vào một chuỗi để in ra trang web • Nhập vào ký tự ”q” để thoát vòng lặp* |  |
| cú pháp vòng lặp do-while |  |
| Điểm khác biệt giữa while và do-while | cau lenh while khong xac dinh xets dieu kien xong moi lawp  while do khong han che lawp xong moi xet dk |
| Giải thích được cách sử dụng lệnh break | Lệnh break dùng để thoát khỏi vòng lặp mà không đi tới điểm cuối |
| Giải thích được cách sử dụng lệnh continue | bo qua 1 buoc tiep tuc cau lenh |
| So sánh được các loại vòng lặp |  |
| Giải thích vòng lặp lồng nhau | Demo code sinh bảng cửu chương |
| Phân biệt các tình huống sử dụng vòng lặp khác nhau |  |
| Lặp vô hạn | Vòng lặp vô hạn là chuỗi các câu lệnh được thực hiện lặp đi lặp lại không có điểm dừng |
|  |  |
|  |  |
| **Mảng** | |
|  | mang la bien dac biet chuwa nhieu gia tri  c1: let arr = [element1, element2, element3]  c2: let array =new Array(element1, element2, element3)  Hàm push() sẽ thêm một phần tử mới vào phần cuối của mảng.  cars.push("Kia")  Hàm unshift() sẽ thêm một phần tử mới và phần đầu của mảng.  cars.push("Kia")  Hàm pop() sẽ lấy đi phần tử cuối cùng của mảng.  let last = cars.pop() // Kia  Hàm shift() sẽ lấy đi phần tử đầu tiên của mảng.  let last = cars.shift() // "Ferari"  Hàm sort() sẽ giúp sắp xếp các phần tử của mảng theo một trật tự nhất định.  Chúng ta cũng có thể đảo ngược trật tự của một mảng bằng cách sử dụng hàm reverse().  let arr = ["Toyota", "Subaru", "BMW"]  arr.sort() // [ "BMW", "Subaru", "Toyota" ]  arr.reverse() // [ "Toyota", "Subaru", "BMW" ]        for in duyet chi so  for of duyet ptu |
| **Hàm** | |
| Hàm là gì ? | - |
| Cú pháp khai báo hàm không tham số | - |
| Quy tắc khai báo hàm |  |
| Cú pháp khai báo hàm có tham số |  |
| Cú pháp khai báo hàm có tham số mặc định |  |
| Sự trả về của hàm |  |
| Tham chiếu và tham trị |  |
| **Lập trình hướng đối tượng** | |
| Lập trình HDT là gì ? | - |
| Đối tượng là gì ? | - |
| Lớp là gì ? | - |
| Thuộc tính là gì ? | - |
| Phương thức là gì ? | - |
| Cách khởi tạo 1 đối tượng |  |
| Cách khai báo 1 lớp |  |
| Khởi tạo đối tượng từ lớp |  |
| Sử dụng thuộc tính |  |
| Sử dụng phương thức |  |
| Từ khóa "this" dùng để làm gì? | bạn đã sử dụng dấu chấm (".") để truy xuất các thuộc tính và phương thức của đối tượng  Cặp ngoặc vuông ("[]")  Để truy xuất các thuộc tính của đối tượng bạn có thể dụng cặp ngoặc vuông ("[]") thay cho dấu chấm ("."):  Từ khóa this dùng để tham chiếu đến đối tượng hiện hành - |
| Sơ đồ mô tả lớp |  |
| Các tính chất của OOP |  |
| Vẽ hình tròn canvas |  |
| Vẽ HCN canvas |  |
| Vẽ chữ canvas |  |