**Pràctica 9**

**Estructura repetitiva (while)**

Fins ara hem emprat estructures SEQÜENCIALS i CONDICIONALS. Hi ha un altre tipus d'estructures tan importants com les anteriors que són les estructures REPETITIVES.  
Una estructura repetitiva permet executar una instrucció o un conjunt d'instruccions diverses vegades.  
  
Una execució repetitiva de sentències es caracteritza per:  
- Les sentències que es repeteixen.  
- El test o prova de condició abans de cada repetició, que motivarà que es repeteixin o no les sentències.  
  
Funcionament del ‘while()’: En primer lloc es verifica la condició, si la mateixa resulta vertadera, s'executen les operacions que indiquem entre les claus que li segueixen al ‘while()’.  
En cas que la condició sigui falsa contínua amb la instrucció següent al bloc de claus.  
El bloc es repeteix MENTRE la condició sigui Vertadera.  
  
Important: Si la condició sempre retorna vertader estem en presència d'un cicle repetitiu infinit. Aquesta situació és un error de programació, mai finalitzarà el programa.  
  
Exemple: Realitzar un programa que imprimeixi en pantalla els números de l'1 al 100.  
  
<!DOCTYPE html>

<html>

<head>

<title>Pràctica 9</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<script>

function inici(){

var x;

x = 1;

document.write('<h1>Números del 1-100</h1><br>');

while (x <= 100){

document.write(x);

document.write('<br>');

x = x + 1;

}

}

window.onload=inici;

</script>

</head>

<body>

</body>

</html>

Sense conèixer les estructures repetitives podem resoldre el problema emprant una estructura seqüencial. Inicialitzem una variable amb el valor 1, després imprimim la variable, incrementem novament la variable i així successivament.

És molt important analitzar aquest programa:  
La primera operació inicialitza la variable ‘x’ a 1, seguidament comença l'estructura repetitiva while i disposem la següent condició (x <= 100), es llegeix MENTRE la variable x sigui menor o igual a 100.  
En executar la condició, retorna CERT, perquè el contingut de ‘x’ (1) és menor o igual a 100. Com que la condició es vertadera, s'executa el bloc d'instruccions que conté l'estructura ‘while’. El bloc d'instruccions conté dues sortides al document i una operació. S'imprimeix el contingut de ‘x’ i seguidament s'incrementa la variable ‘x’ en un.  
  
L'operació x = x + 1 es llegeix com "en la variable ‘x’ es guarda el contingut de ‘x’ més 1". És a dir, si x conté 1 després d'executar-se aquesta operació s'emmagatzemarà 2.  
  
En finalitzar el bloc d'instruccions que conté l'estructura repetitiva, es verifica novament la condició de l'estructura repetitiva i es repeteix el procés explicat anteriorment.  
Mentre la condició retorni vertader, s'executa el bloc d'instruccions; en retornar fals la verificació de la condició, es surt de l'estructura repetitiva i continua l'algoritme, en aquest cas, finalitza el programa.  
  
El més difícil és la definició de la condició de l'estructura ‘while’ i quin bloc d'instruccions es repetirà. Observar que si, per exemple, disposem la condició x >= 100 (si ‘x’ és més gran o igual a 100) no provoca cap error sintàctic però estem en presència d'un error lògic perquè al avaluar-se per primera vegada la condició retorna fals i no s’executa el bloc d'instruccions que volíem repetir 100 vegades.  
  
No hi ha una RECEPTA per definir una condició d'una estructura repetitiva, sinó que s'aconsegueix amb una pràctica contínua, solucionant problemes.  
  
Quan ‘x’ val 101 la condició de l'estructura repetitiva retorna fals, en aquest cas finalitza el diagrama.

La variable ‘x’ rep el nom de COMPTADOR. Un comptador és un tipus especial de variable que s'incrementa o decrementa amb valors constants durant l'execució del programa. El comptador x ens indica en cada moment la quantitat de valors impresos a la pàgina.

Important: Podem observar que el bloc repetitiu pot no executar-se si la condició retorna fals la primera vegada.

La variable ‘x’ ha d'estar inicialitzada amb algun valor abans que s'executi l'operació x = x + 1.

**PROBLEMES**

1. Realitzar un programa que imprimeixi 25 termes de la sèrie 11-22-33-44, etc. (No s'ingressen valors per teclat).
2. Mostra els múltiples de 8 fins al valor 500. Ha d'aparèixer en pantalla 8-16-24, etc.

**Concepte d'acumulador**  
  
Explicarem el concepte d'un acumulador amb un exemple.

Problema: Desenvolupar un programa que permeti la càrrega de 5 valors per teclat i ens mostri posteriorment la suma.

<html>

<head>

<title>Pràctica 9</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<script>

function inici(){

var x = 1;

var suma = 0;

var valor;

while (x <= 5){

valor = parseInt(prompt('Introduïu un valor:'));

suma = suma + valor;

x = x + 1;

}

document.write("La suma dels valors és: " + suma + "<br>");

}

window.onload=inici;

</script>

</head>

<body>

</body>

</html>

En aquest problema, a semblança dels anteriors, portem un COMPTADOR anomenat ‘x’ que ens serveix per comptar les voltes que ha de repetir el ‘while’.  
També apareix el concepte d’ACUMULADOR (un acumulador és un tipus especial de variable que s'incrementa o decrementa amb valors variables durant l'execució del programa).  
Hem donat el nom de suma al nostre acumulador. Cada cicle, la variable suma s'incrementa amb el contingut ingressat a la variable valor.

**PROBLEMES**

1. Escriure un programa que llegeixi 10 notes d’alumnes i ens informe quants tenen notes majors o iguals a 7 i quants menors.
2. S’ingressen un conjunt de 5 altures de persones per teclat. Mostra l'altura mitjana de les persones.
3. En una empresa treballen 5 empleats. Els sous oscil·len entre $ 1000 i $ 5000, realitzar un programa que llegeixi els sous que cobra cada empleat i informe quants empleats cobren entre $ 1000 i $ 3000 i quants cobren més de $ 3000. A més el programa haurà d'informar l'import que gasta l'empresa en sous al personal.
4. Realitzar un programa que imprimeixi 20 termes de la sèrie 5-10-15-20, etc. (No s'ingressen valors per teclat).
5. Mostrar els múltiples de 10 fins al valor 1500.  
   Ha d'aparèixer en pantalla 10-20-30 etc.
6. Desenvolupar un programa que permeti carregar 5 números sencers i després ens informi quants valors són parells i quants imparells.  
   Emprar l'operador "%" a la condició de l'estructura condicional.  
   if (valor % 2 == 0)  
   L'operador "%" retorna la resta de la divisió (valor / 2). Per exemple: 12 % 2, retorna 0; 13% 2, retorna 1, perquè la resta de dividir 13 a 2 és 1.