

# TP et Labo de la semaine 12 + solutions

## Opérations sur les tableaux

1) Utilisez la méthode map sur les tableaux pour faire les traitements suivants :

- diviser par 2 chaque élément d'un tableau de nombres
- multiplier par n chaque élément d'un tableau de nombres
- créer une matrice 10x10 de nombres tel que le nombre en position (i,j) est égal à  $i*j$ , donc la matrice est une table de multiplication

Solution :

```
var moitie = function (x) {
    return x/2;
};
var tabDiv2 = function (tab) {
    return tab.map(moitie);
};
var tabDiv2 = function (tab) { // autre solution
    return tab.map(function (x) { return x/2; });
};
var tabMulN = function (tab, n) {
    return tab.map(function (x) { return x*n; });
};
/**
 * Retourne un tableau de nombres croissants allant de 0 à n (exclus)
 */
var iota = function ( n ) {
    return Array( n ).fill(0).map( function (x,i) { return i; } );
};
var tableDeMult = function ( n ) {
    var nbs = iota( n );
    return nbs.map(function (x) { return tabMulN(nbs, x); });
};
var mat = tableDeMult(10);
```

2) Étant donné un tableau d'enregistrements contenant un champ id et un champ notes, tel que vu au cours pour la fusion de notes, utilisez la méthode map sur les tableaux pour faire les traitements suivants :

- créer un tableau de tous les identifiants
- créer un tableau de toutes les notes du Nieme travail (n=0, 1, ...)

Solution :

```
var identifiants = function (tab) {
    return tab.map(function (rec) { return rec.id; });
};
var notesDuTravailN = function (tab, n) {
    return tab.map(function (rec) { return rec.notes[n]; });
};
```

3) Écrire une fonction "somme" avec la méthode `reduce` qui prend un tableau de nombres en paramètre et qui calcule la somme de ces nombres.

```
var somme = function(tab) {  
    return tab.reduce(function(acc, x) {  
        return acc + x;  
    }, 0);  
};
```

4) Écrire une fonction "produit" avec ta méthode `reduce` qui prend un tableau de nombre en paramètre et qui calcule le produit de ces nombres.

```
var produit = function(tab) {  
    return tab.reduce(function(acc, x) {  
        return acc * x;  
    }, 1);  
};
```

5) Écrire une fonction "maximum" avec la méthode `reduce` qui prend un tableau non-vide de nombres en paramètre et qui retourne la plus grande valeur du tableau.

```
var max2 = function(a, b) {  
    return a > b ? a : b;  
};  
var maximum = function(tab) {  
    return tab.reduce(max2, -Infinity);  
};
```

6) Écrire une fonction "myJoin" avec la méthode `reduce` qui prend en paramètre un tableau de strings et un élément avec lequel joindre chaque élément du tableau pour

produire une string.

```
var myJoin = function(tab, separateur) {  
    if(tab.length == 0)  
        return "";  
    return tab.splice(1).reduce(function(acc, x) {  
        return acc + separateur + x;  
    }, tab[0]);  
};
```

### Manipulation de fichiers

1) Écrire la fonction `sommeFichier(path)` qui retourne la somme de tous les nombres dans le fichier ayant le chemin d'accès `path`. Le fichier contient uniquement des nombres (0-9), des espaces et des fins de ligne.

Solution :

```
var sommeFichier = function (path) {  
  
    var contenu = readFile(path);  
    // retirer sauts de ligne  
    var sansSauts = contenu.split("\n").join(" ");  
    var nombres = sansSauts.split(" "); // extraire nombres  
    var s = 0;
```

```

    for (var i=0; i<nombres.length; i++) {
        if (nombres[i] != "") { // traiter plusieurs espaces consécutifs
            s += +nombres[i];
        }
    }
    return s;
};

```

2) Le fichier "bottin.csv" est un fichier CSV qui contient un bottin téléphonique associant à chaque nom un numéro de téléphone. Voici un exemple de fichier "bottin.csv":

```

jean,450 831-5133
luc,514 824-9913
marie,819 123-4567

```

On cherche à écrire un programme interactif qui va permettre de consulter le bottin, soit en cherchant un nom (pour trouver le numéro de téléphone de la personne) ou un numéro de téléphone (pour trouver le nom de la personne), ou bien pour ajouter des personnes au bottin. Le programme doit utiliser la fonction "prompt" pour demander la commande à l'utilisateur, et pour afficher le résultat. La commande "fin" ou une ligne vide doit terminer le programme. Lorsque le programme termine le nouveau fichier "bottin.csv" doit être sauvé.

Solution :

```

var lettre = function (car) {
    return (car >= "a" && car <= "z") ||
           (car >= "A" && car <= "Z");
};

var lireCSV = function (path) {
    var lignes = readFile(path).split("\n");
    var resultat = [];
    for (var i=0; i<lignes.length; i++) {
        resultat.push(lignes[i].split(","));
    }
    return resultat;
};

var ecrireCSV = function (path, matrice) {
    var contenu = [];
    for (var i=0; i<matrice.length; i++) {
        contenu.push(matrice[i].join(","));
    }
    writeFile(path, contenu.join("\n"));
};

var positionNom = function (bottin, nom) {
    for (var i=0; i<bottin.length; i++) {
        if (bottin[i][0] == nom) {
            return i;
        }
    }
    return -1;
};

```

```

var positionNumero = function (bottin, numero) {
    for (var i=0; i<bottin.length; i++) {
        if (bottin[i][1] == numero) {
            return i;
        }
    }
    return -1;
};

var gererBottin = function () {

    var bottin = lireCSV("bottin.csv");

    while (true) {
        var cmd =
            prompt("Entrez un nom, numéro de téléphone ou 'fin' pour quitter");
        if (cmd == "fin" || cmd == "") {
            break;
        }

        if (lettre(cmd.charAt(0))) {
            // entré un nom... le chercher
            var pos = positionNom(bottin, cmd);

            if (pos == -1) {
                var numero =
                    prompt("Nom inconnu. Entrez le numéro du contact à ajouter :");
                bottin.push([cmd, numero]);
            }
            else {
                alert(bottin[pos].join(" : "));
            }
        } else
            if (cmd.charAt(0) >= "0" && cmd.charAt(0) <= "9") {
                // entré un numéro de téléphone... le chercher
                var pos = positionNumero(bottin, cmd);

                if (pos == -1) {
                    var nom =
                        prompt(
                            "Numéro inconnu. Entrez le nom du contact à ajouter :");
                    bottin.push([nom, cmd]);
                } else {
                    alert(bottin[pos].join(" : "));
                }
            }

    }

    ecrireCSV("bottin.csv", bottin);
};

gererBottin();

```