

AUCUNE DOCUMENTATION PERMISE

- (10) 1. Parmi les types simples XML prédéfinis, on trouve les types **ID** et **IDREF**. Expliquez leur comportement et pourquoi ils sont utiles lors de la validation d'un fichier XML.
- (20) 2. Vous allez vous constituer un fichier XML de notices bibliographiques avec les contraintes suivantes:
- La bibliographie comprend une suite de notices sur des **articles**, des **conférences** ou des **livres**, et ce dans n'importe quel ordre.
 - Toutes les notices ont un attribut **année**.
 - Les auteurs sont donnés par une suite d'éléments qui donnent séparément le prénom et le nom d'un auteur sous forme de chaînes de caractères.
 - les numéros de pages sont donnés par une ou deux suites de chiffres séparées par un tiret; p.ex. 23 ou 123-145.
 - Un **article** comprend les informations suivantes dans l'ordre:
 - ses auteurs
 - le titre et le nom du journal sous forme d'une chaîne de caractères
 - un numéro de volume (un entier) optionnel
 - les numéros de pages
 - une conférence est décrite par les éléments suivants dans l'ordre
 - ses auteurs
 - son titre
 - le nom de la conférence
 - un livre est décrit de la façon suivante dans l'ordre
 - ses auteurs
 - son titre
 - son nombre de pages (un entier)

Complétez le fichier de validation RelaxNG (forme compacte) suivant pour implanter ces contraintes:

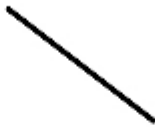
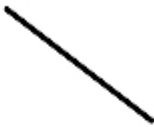
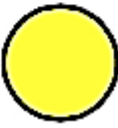
```
datatypes xsd = "http://www.w3.org/2001/XMLSchema-datatypes"
start = element bibliographie {
    # à compléter...
}
```

(20) 3. Soit une page web décrite par le code HTML suivant:

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Prédisez le résultat</title>
  </head>
  <body>
    <h1>Contactez-nous</h1>
    <form action="http://www.iro.umontreal.ca/ift1005/contact.cgi" method="get">
      <fieldset>
        <legend>Merci de remplir ce formulaire et on vous contactera</legend>
        <div>Courriel:
          <input type="email" name="e-mail" size="40"
            placeholder="adresse courriel"/>
        </div>
        <div>
          Indiquez ce qui vous intéresse:
          <select name="service" size="1">
            <option value="dev">Développement web</option>
            <option value="des">Design web</option>
            <option value="ent">Entretien web</option>
            <option value="info">Information générale</option>
          </select>
        </div>
        <div>
          <input type="submit" value="Envoyer" id="OK"/>
        </div>
      </fieldset>
    </form>
    <nav>
      <a href="index.html">Accueil</a>
      <a href="services.html">Services</a>
      <a href="contact.html">Contact</a>
    </nav>
  </body>
</html>
```

(10) (a) Dessinez ce qu'affichera cette web et décrivez en les éléments.

(10) (b) Décrivez ce qui arrivera si quelqu'un clique sur l'élément de la page identifié par input#OK.

Formes SVG dans un carré unitaire		
Description	Forme de base	Paths
Ligne de 0.1,0.3 à 0.75,0.8		
Triangle sommets: 0.5,0.2 0.2,0.8 0.9,0.7		
Cercle rayon:0.25, centre:0.4,0.6		

- (20) 4. Écrivez le code HTML et la feuille de style correspondante qui permet de reproduire la figure ci-haut qui décrit des formes SVG à dessiner. Chaque case dans les deux dernières colonnes du tableau sont des éléments SVG de 3cm par 3cm qui correspondent à un carré unitaire dans l'espace de dessin, c.-à-d. les coordonnées x et y sont comprises dans l'intervalle $[0,1]$. Les lignes sont noires, d'une largeur de trait de 0.02. Le remplissage des figures est en jaune. Dans la deuxième colonne du tableau, les figures doivent être produites avec une forme de base SVG; dans la troisième colonne, il faut produire la figure équivalente, mais en utilisant plutôt l'élément SVG `path`.
- (10) 5. Pourquoi est-ce que la question de la crédibilité (*trust* en anglais) est-elle importante dans le monde du web? Décrivez trois moyens pour améliorer la crédibilité des informations sur un site web.

- (10) 6. Décrivez le comportement de la fonction associée au champ `lambda` (un nom volontairement vague dans cet examen) dans cet objet Javascript. Donnez trois exemples d'appels qui illustrent son comportement.

```
var obj = {  
  lambda: function (c, val) {  
    if (this[c]) {  
      if (typeof(this[c]) == "object") {  
        this[c].push(val)  
      } else {  
        this[c] = [this[c], val]  
      }  
    } else {  
      this[c] = val;  
    }  
  }  
}
```

- (10) 7. Définissez ce que sont les sélections en D3, comment les crée-t-on et que permettent-elles de faire dans une page HTML.

Cet examen comporte 7 questions pour un total de 100 points.

Bonne chance

Rappels RelaxNG compact

id = *pattern*

Sortes de *patterns*

element nom { *pattern* }

attribute nom { *pattern* }

mixed { *pattern* }

2 *patterns* combinés avec , ou & ou |

pattern suivi de ? ou * ou +

id

text

Rappels HTML

Structure d'une page

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"/>
  <link href="..." type="stylesheet"/>
  <title>...</title>
</head>
<body>
  ...
</body>
</html>
```

Éléments du body

em,strong	mise en relief
sub,sup	indice, exposant
ul	liste non-ordonnée
ol	liste ordonnée
li	élément de liste
div,span	regroupements d'éléments
a href="..."	lien entre documents
img src="..." alt="..."	image
table	tableau
caption	légende d'un tableau
tr	ligne d'un tableau
th	case d'entête de tableau
td	case de tableau

Éléments de division HTML5

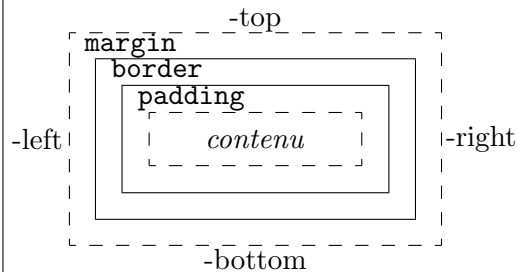
header	entête de toute la page
nav	navigation principale
aside	information auxiliaire
section	contenu de la page
article	unité d'information autonome
footer	pied de page

Rappels CSS

Sélecteurs

<i>nom</i>	nom d'un élément
<i>.classe</i>	nom d'une classe
<i>#id</i>	identificateur d'un élément
<i>:pseudo-classe</i>	:link, :hover, :active

Modèle de boîte



Principales propriétés

background-color	couleur du fond
background-image	url(<i>image</i>)
background-repeat	repeat-x repeat-y no-repeat
background-position	left top center bottom right
border	border-width border-style <i>couleur</i>
color	spécification de la couleur
display	inline, block
font	font-family font-style font-weight
height	hauteur du bloc
margin	largeur de la marge
padding	largeur du <i>padding</i>
position	relative absolute fixed
text-align	left right center justify
top	position en haut à gauche
width	largeur du bloc

[illegible]

d3 cheat sheet

Select > data > enter > append :

select	CSS selector (string)	Selects the <i>first</i> element that matches the selector.
selectAll	CSS selector (string)	Selects <i>all</i> the elements that match the selector.
Selections can be chained : each new statement will look <i>inside</i> the current selection.		
data	array of items	Provides the data that will be matched against the selection
enter	nothing - ()	Selects elements to be added: the items in the data which are not yet matched by elements.
exit	Nothing - ()	Selects elements to be removed: the elements which are no longer matched by items in the data
append	name of element	creates specified element inside the selection, either once (if data/enter is not used) or as many times as there are elements in enter().
remove	Nothing - ()	removes selected elements.

The following constructs are valid.

d3.select("body").append("svg");	Creates an svg element in the HTML document (assuming there is a body element)
var myRects=d3.select("body").select("svg").selectAll("rect");	Looks for all rect elements within the first svg element found within the body element.
myRects=myRects.data([1,2,3]).enter();	If there are fewer than 3 rect elements in myRects, this prepares the missing elements to be added.
myRects.append("rect");	Create these missing elements. Extra methods can be added to initiate them (attr, style...) will only apply to the elements just created
d3.selectAll("rect");	This selects all rect elements. Methods added there will apply to all of the rect elements, not only to those just created.
d3.selectAll("rect").data([4,5]).exit().remove();	This passes new data to the rect elements. Then, if there are more than 2 of them, the rest – exit() – are deleted – remove().
d3.selectAll("rect").remove();	This deletes all rect elements.

The following are not:

d3.append("svg");	All elements must be added within a container element. <i>First</i> select, <i>then</i> append.
d3.selectAll("rect").enter().append("rect"); d3.data([1,2,3]).enter().append("rect");	enter() requires <i>first</i> a selection, <i>then</i> data.
d3.select("rect").data([1,2,3]).enter().append("rect");	To use data and enter, you need to use selectAll first. (else elements are created, but not where you'd expect.)

Path – what goes in the d attribute:

M	x,y	Begins the string. M moves the cursor to the designated position to draw a shape. If more than one pair of coordinates are provided, it's as if the other pairs are preceded by an L.
m	x,y	Same as above, the only difference is that if more than one pair of coordinates are provided, extra pairs are processed as if preceded by an l.
L	x,y	Draws a line from the last position to the specified position.
l	x,y	Draws a line from the last position to a relative position: x pixels to the right and y to the bottom.
H	x	Draws a horizontal line to the specified, absolute x position (y stays the same)
h	x	Draws a horizontal line x pixels to the right (if x is negative, the line is drawn to the left).
V	y	Draws a vertical line to the specified, absolute y position (x stays the same)
v	y	Draws a vertical line x pixels to the bottom (if y is negative, the line is drawn to the top).
A	rx,ry,alpha,large,sweep,x,y	Draws an elliptical arc: rx and ry are the radius of the ellipse; alpha is the x-axis rotation of the ellipse; large is 0 if the arc should be the shorter arc (less than 180°), 1 if it should be the longer arc; sweep is 0 if the arc is to be drawn clockwise, 1 if counter-clockwise; x,y are the coordinate of the end point of the arc.
Q	cX cY eX eY	Draws a Bézier quadratic curve. cX,cY are the coordinates of the control point, eX,eY that of the endpoint. More pairs can be provided (extra control points and endpoints).
q	cX cY eX eY	Draws a Bézier quadratic curve, with the coordinates of the points relative to the current point.
T	eX eY	Draws a Bézier quadratic curve, using the last provided control point (or failing that the current point).
t	eX eY	Same as above, with the coordinates of the end point relative to the current point.
C	cX1 cY1 cX2 cY2 eX eY	Draws a Bézier cubic curve, with cX1,cY1, cX2, cY2 being the coordinates of the control points.
c	cX1 cY1 cX2 cY2 eX eY	Same as above, with the coordinates of the points relative to the current point.
S	cX2 cY2 eX eY	Draws a Bézier cubic curve, using the previously provided control point (cX1,cY1) or failing that the current point, cX2,cY2 as the next control point, and eX, eY as the end point
s	cX2 cY2 eX eY	Same as above, with the coordinates of the points relative to the current point.
Z		Optional at the end the string for a closed shape (ie line to the first point)