

CORRECTION du DS de synthèse de la SERIE 2 de TP

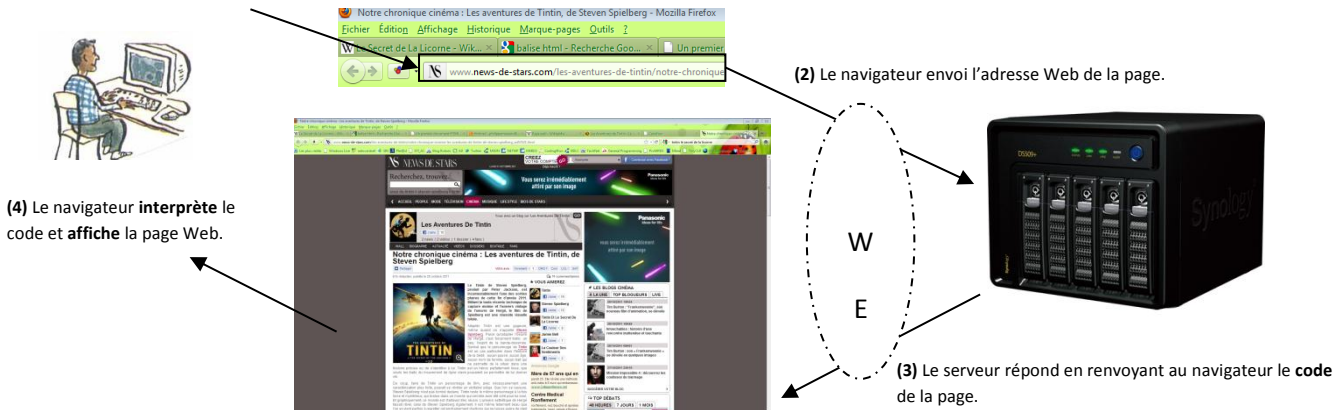
Les réponses ci-dessous sont dans le sujet du TP1 « Site Web »

Q1)

(1) L'utilisateur écrit l'adresse Web (URL) :

Navigateur Web (client)

Serveur Web



Q2) Un navigateur Web est un logiciel conçu pour consulter le World Wide Web. Il utilise le protocole HTTP.

Q3) Le HTML (« Hypertext Mark-Up Language») est un langage dit de « marquage » (de "structuration" ou de "balisage") dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage.

Q4) Une balise est un élément de texte (un nom) encadrée par le caractère inférieur ("<") et le caractère supérieur (">")

Q5)



Le site Web GE a été créé en 2006 par des élèves de terminale S avec **Dreamweaver8**

- **Transformation automatique**

On peut obtenir une page web par transformation automatique d'un document issu d'un traitement de texte, d'un tableur, . . . Le résultat est souvent très « **lourd** » et rarement totalement conforme aux standards du W3C. Mais c'est une solution simple et rapide.

- **Utilisation d'un éditeur web WYSIWYG**

Un éditeur web **WYSIWYG** (**What you see is what you get**) est un logiciel de création de page web (Quanta, **BlueGriffon**, Nvu, **Macromedia Dreamweaver**, Microsoft FrontPage, Amaya etc.).

- Un éditeur de texte simple (Bloc-note, Notepad) ou un éditeur de code proposant la coloration syntaxique et l'auto complétion (Notepad++, PSPad, SciTE).

Les réponses ci-dessous sont dans le document réponse du TP1 « Site Web »

Q6)

- **Règle 1** : Ecrire les balises en **minuscule**.
- **Règle 2** : Toutes les balises **ouvertes** par `<nom_balise>` doivent être **fermées** par `</nom_balise>` sauf s'il s'agit d'une balise unique `<balise_unique />`.
- **Règle 3** : Les balises doivent être correctement **imbriquées** et **indentées**
- **Règle 4** : Les valeurs des attributs doivent toujours figurer entre des **guillemets**.

Les réponses ci-dessous sont dans le sujet des TP4 et TP5 « NXT »

Q7) Dialogue entre le robot NXT et le PC

Le contrôle du robot s'effectue par l'intermédiaire d'une liaison **Bluetooth**. Le programme n'est pas transféré dans le robot mais **s'exécute sur le PC**.



Q8) Principe adopté pour le déplacement du robot : Conduite différentielle (Differential drive)

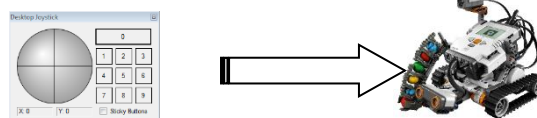
Conduite différentielle !

Q9) Modes de déplacement

Nous retenons deux modes de fonctionnement : déplacement piloté ou déplacement autonome.

Mode piloté

Dans ce mode, un pilote contrôle le robot par l'intermédiaire d'une commande (joystick.)



Mode autonome

Le robot se déplace seul en respectant une séquence de déplacements élémentaires (ligne droite, virage).

Lors du déplacement du robot, il peut être nécessaire de contrôler deux paramètres :

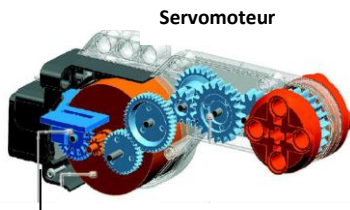
- la **distance** parcourue,
- la **vitesse** de déplacement.

Q10) Mesure du déplacement du robot par odométrie

Odométrie !

L'odométrie est une technique permettant d'**estimer** la position d'un véhicule en **mouvement**. Cette mesure de **bas niveau** est présente sur quasiment tous les robots mobiles, grâce à des **capteurs de rotation**.

Sur le **NXT**, le **capteur de rotation intégré au servomoteur**, mesure la rotation du moteur en degrés (précision **+/-1 degré**)



Capteur de rotation intégré

L'odométrie repose sur la **mesure individuelle des déplacements des roues** pour reconstituer le mouvement global du robot. En partant d'une position initiale connue et en ajoutant les déplacements mesurés, on peut ainsi calculer à chaque instant la position courante du véhicule. Dans le logiciel Microsoft Robotics Developer Studio, ce principe est mis en œuvre par le service « **GenericDifferentialDrive** ».

Q11)

Donnée placée dans le programme	« Distance » (m)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Mesure de la distance parcourue	D (cm)	15,5	24,2	33,4	40,9	49,8	59,8	68,7	77	85,5	96
Ecart (cm) = « Distance » – D		-5,5	-4,2	-3,4	-0,9	0,2	0,2	1,3	3	4,5	4

$$0,4 \leq \text{« Distance »}_{(m)} \leq 0,6$$

Les exercices ci-dessous ont été traités dans le TP4 « NXT » et la synthèse des TP

Q12)

Donnée placée dans le programme	« Power »	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Mesure de la distance parcourue	D (cm)	56	57	58,5	59,8	61	61,3	61,5
Ecart (cm) = « Distance » – D		4	3	1,5	0,2	-0.8	-1	-1,5

$$0,7 \leq \text{« Power »} \leq 0,9$$

Q13)

$$0,7 \leq \text{« P »} \leq 0,8$$

Q14) Power (%) = 108.Distance + 13

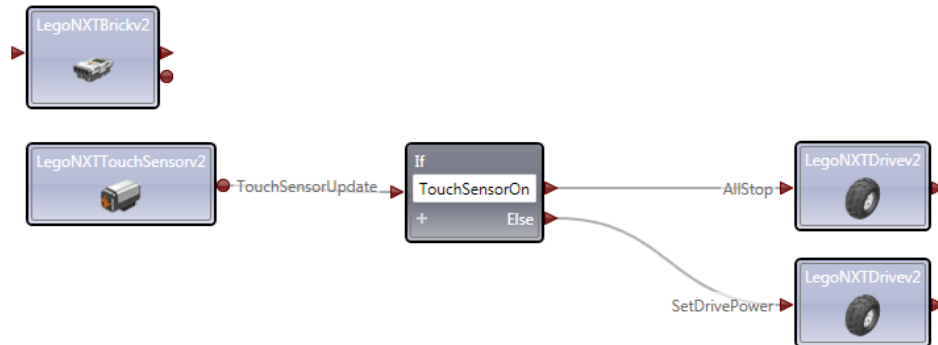
Données placées dans le programme	« Distance » (m)	0.1	0.2	0.4	0.6	0.7
	« Power »	0.24	0.35	0.56	0.78	0.89
Mesure	D (cm)	9,8	19,6	40,5	60,8	71
Ecart (cm) = « Distance » – D		0.2	0.4	-0.5	-0.8	-1

Q15) Oui, $-1 \leq \text{« Ecart »} \leq 1$ sur

$$0,1 \leq \text{« D (m) »} \leq 0,7$$

Les exercices ci-dessous ont été traités dans les TP5, TP6, « NXT »

Q16) {Exercice 1 : TD1 NXT}

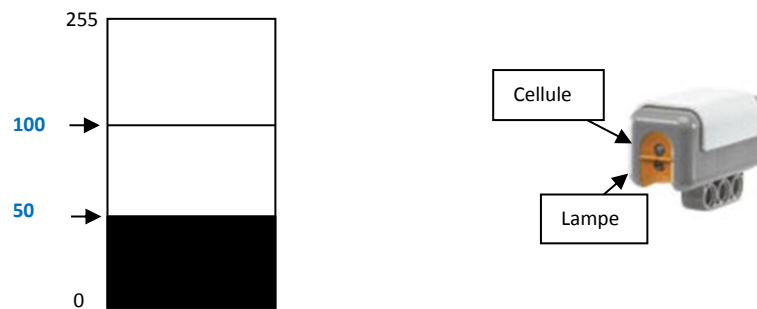


Q17) {TP6 NXT}

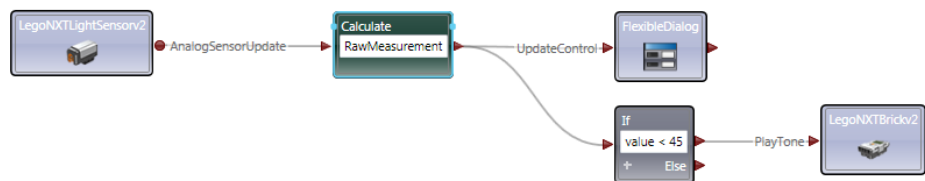
« Lampe » :	« Off »	« On »
	« Luminosité »	
Valeur du « Blanc » :	47	100
Valeur du « Noir » :	21	50
Valeur du « Blanc » - Valeur du « Noir »	26	50

L'écart entre le noir et le blanc (Valeur du « Blanc » - Valeur du « Noir ») étant supérieur lorsque la lampe est « On » dans les conditions de test, on choisit d'utiliser la « Lampe » lors de la détection de la ligne pour mieux différencier le noir du blanc.

Q18) Gabarit avec la lampe « On » pour les conditions de l'essai {TP6 NXT}

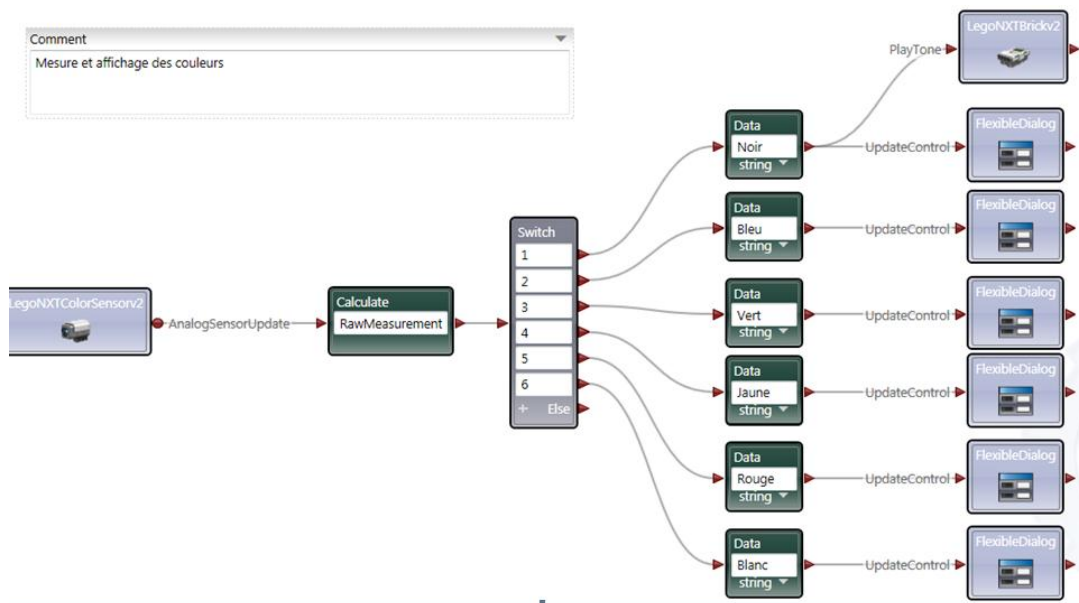


Q19)



Erreur => If (value <45) donne toujours noir.
Correction : Tester value avec une valeur appartenant à]50, 100[.
Par exemple value < 75

Q20)



Algorithme Couleur

début

Lire (le capteur de couleur)
selon (couleur)
1 : Afficher ("Noir") ; fselon ;
2 : Afficher ("Bleu") ; fselon ;
3 : Afficher ("Vert") ; fselon ;
4 : Afficher ("Jaune") ; fselon ;
5 : Afficher ("Rouge") ; fselon ;
6 : Afficher ("Blanc") ; fselon ;
fselon

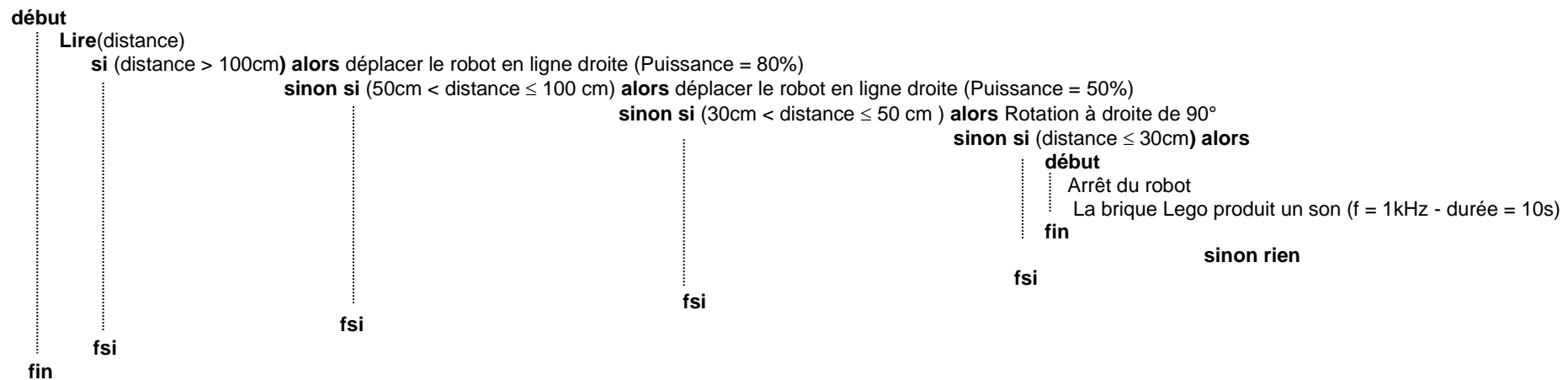
fin

Q21) {Exercice 3 TD1 NXT}

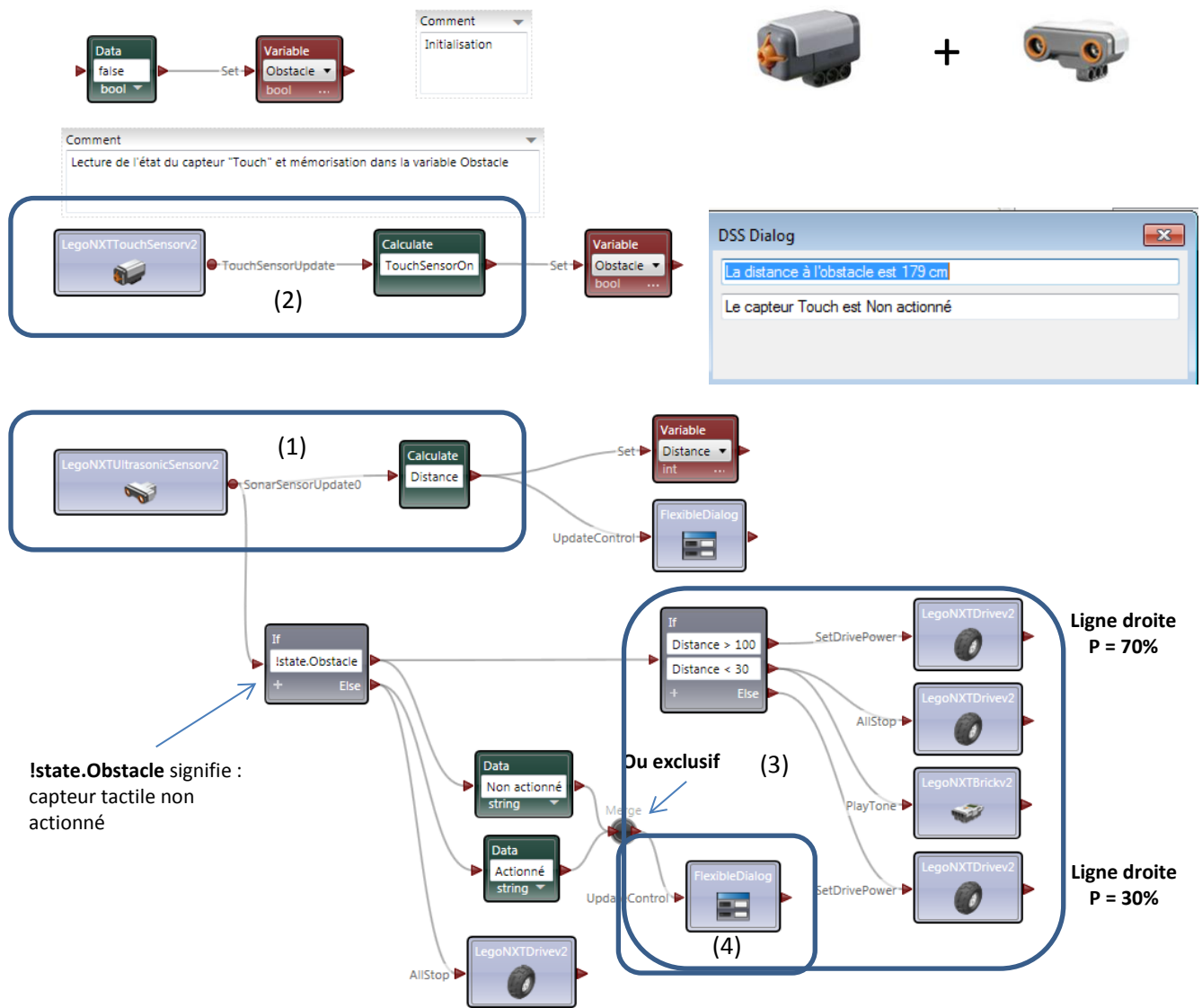
- Distance inférieure ou égale à 100 cm : le robot se déplace en ligne droite avec 50 % de sa puissance
- Distance supérieure à 100 cm : le robot se déplace en ligne droite avec 80% de sa puissance
- Distance inférieure à 30 cm : le robot s'arrête et émet un son (F = 1kHz pendant 10s)



Algorithme Déplacement



On donne le diagramme VPL ci-dessous.



Q22) Entourez :

- (1) La partie du diagramme correspondant à : **Lire**(Distance)
- (2) La partie du diagramme correspondant à : **Lire**(Capteur tactile)
- (3) La partie du diagramme correspondant à :

si (Distance > 100cm) **alors** déplacer le robot en ligne droite (Puissance = 70%)

sinon si (Distance < 30cm) **alors**

début

Arrêt du robot

La brique Lego produit un son

fin

sinon déplacement en ligne droite (Puissance = 30%)

fsi

fsi

- (4) La partie du programme correspondant à : **Ecrire**(l'état du capteur tactile)