

TSTI2D	ALGORITHME - ALGORIGRAMME Arduino	SIN
---------------	--	------------

I. But

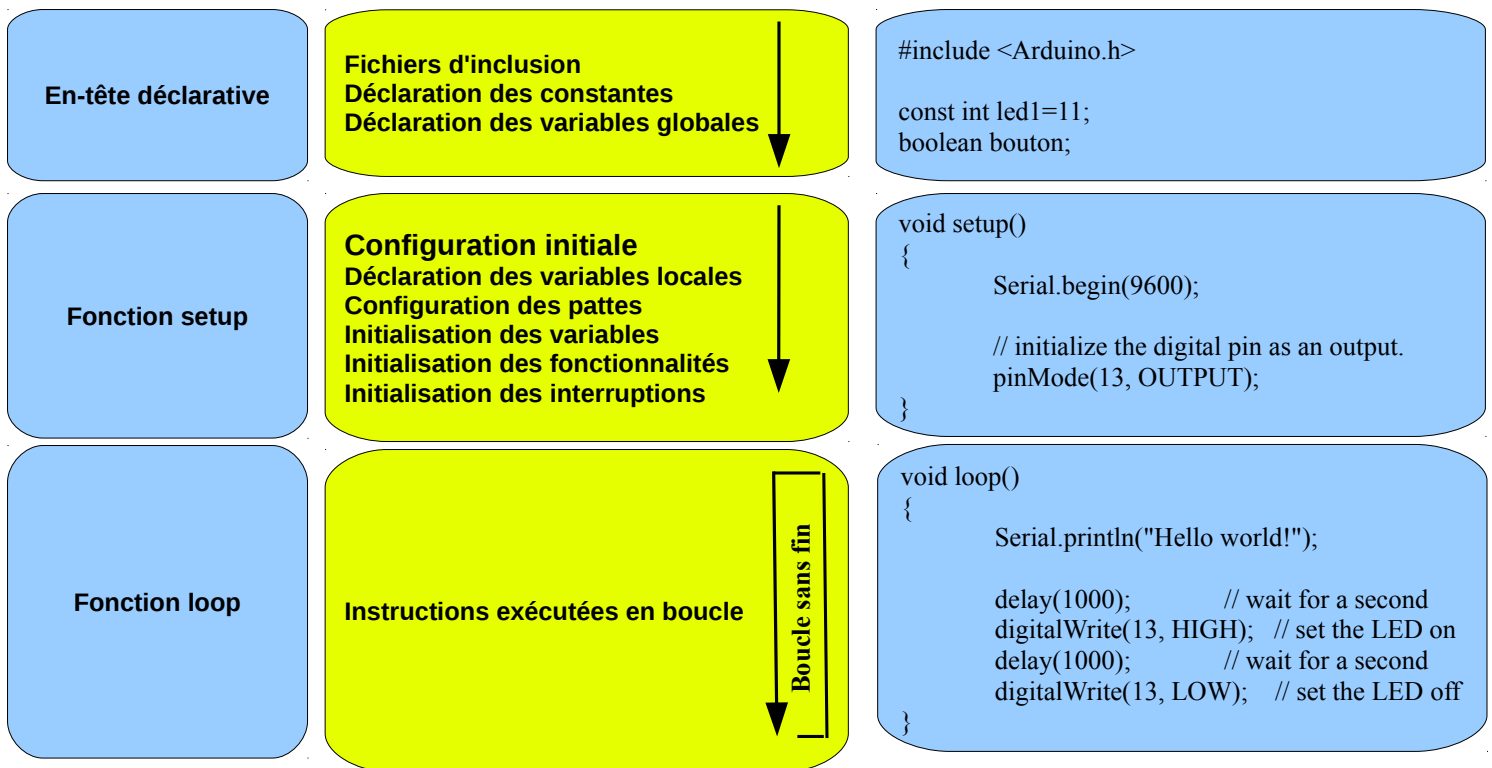
Etude pratique des algorithmes avec arduino.

II. Structure des programmes dans arduino (Rappels)

Un programme dans arduino se déroule de la manière suivante :

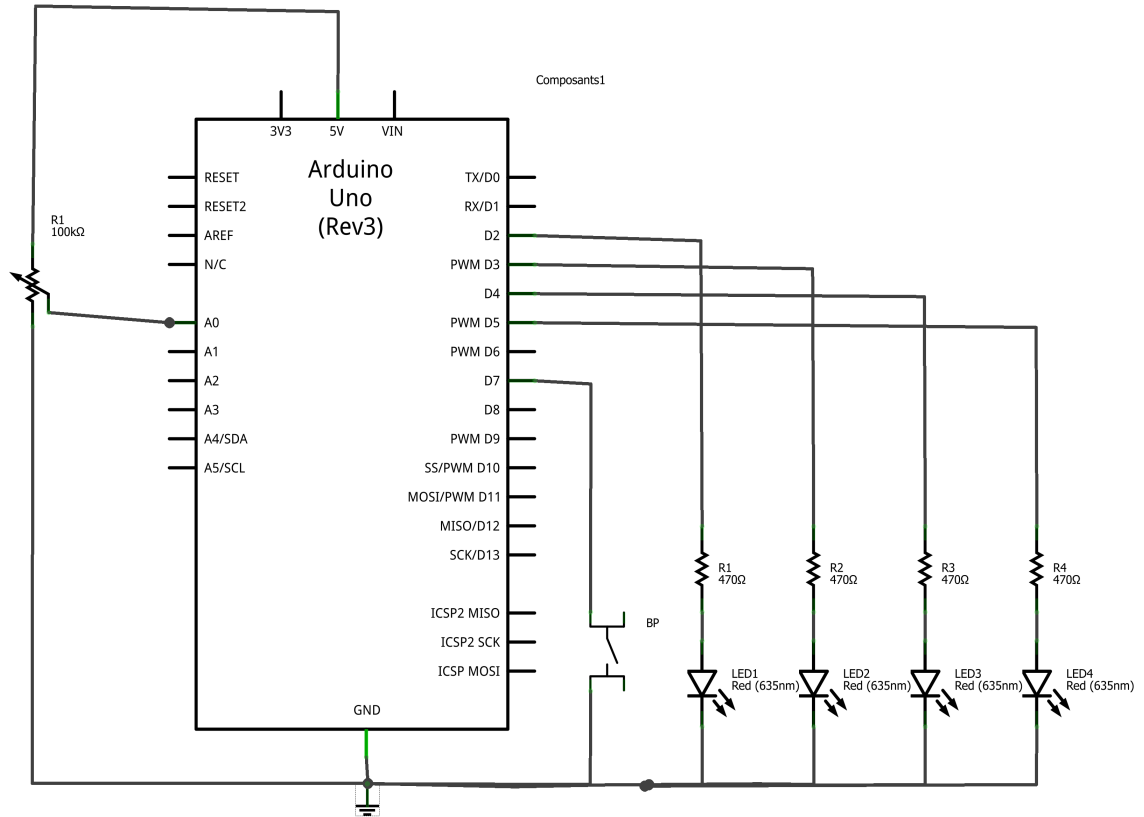
1. Prise en compte des instructions de la partie déclarative
2. Exécution de la partie configuration (*fonction setup()*),
3. Exécution de la boucle sans fin (*fonction loop ()*): le code compris dans la boucle sans fin est exécuté indéfiniment

Déroulement des programmes sur arduino



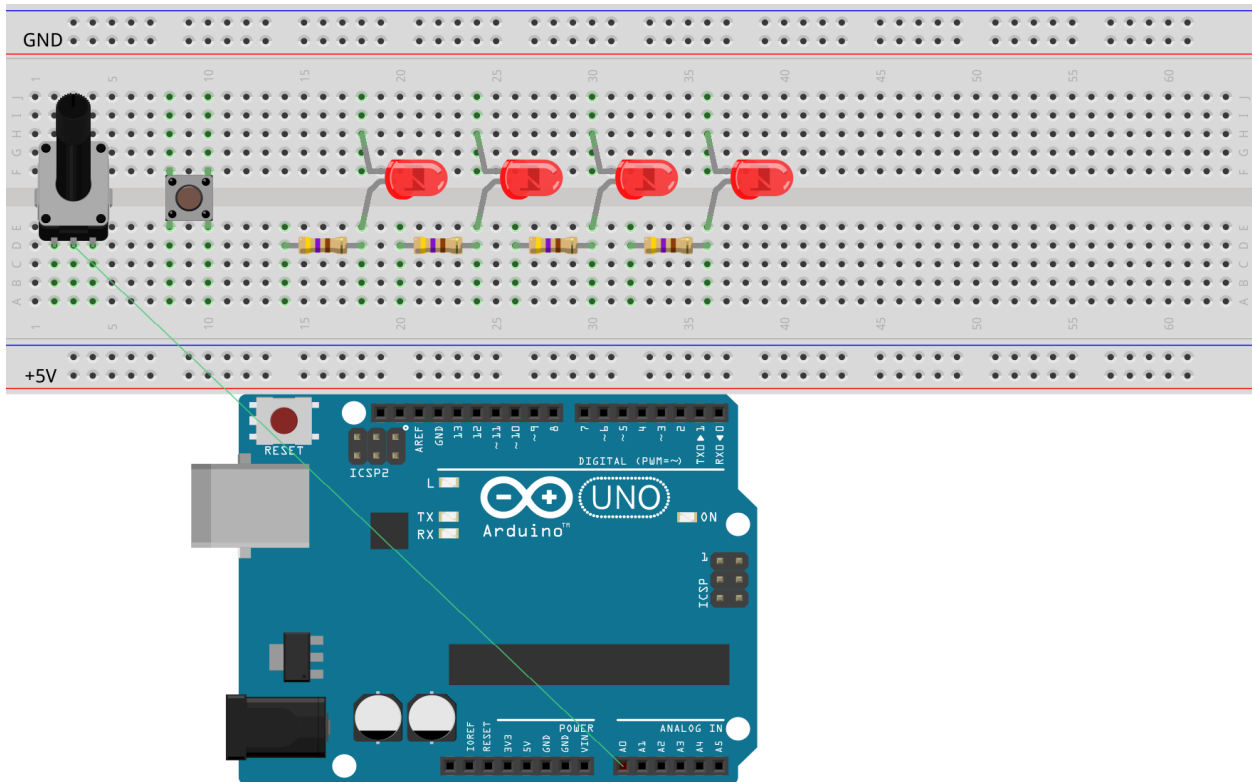
III. Montage

Pour étudier les principales structures des algorithmes, vous allez réaliser le schéma électrique suivant :



Made with Fritzing.org

Compléter le schéma de montage ci-dessous à partir du schéma ci-dessus :



Made with Fritzing.org

Réaliser le schéma ci-dessous.

IV. 1^{er} Algorithme : structure linéaire

Un client souhaite alimenter 4 DEL alternativement..

Une partie de l'algorithme est donné ci-dessous. Compléter le, pour répondre au cahier des charges.

Algorithme	Programme en C++
Constante led1 : entier =2 ~ déclaration de la del 1 ~ ~ déclaration de la del 2 ~ ~ déclaration de la del 3 ~ ~ déclaration de la del 4 ~	const int led1 = 2; //déclaration de la del 1 //déclaration de la del 2 //déclaration de la del 3 ~ //déclaration de la del 4
Procédure setup() Début ~ Configuration initiale ~ initialiser led1 comme une sortie ~ del 1~ ~ del 2~ ~ del 3~ ~ del 4~	void setup() { //Configuration initiale: pinMode(led1, OUTPUT); //del 1 //del 2 //del 3 //del 4 }
Fin	
Procédure loop() Début ~ exécutées en boucle ~ allumer led1 pendant 1 seconde ~ del 1~ éteindre led1 ~ del 1~ ~ del 2~ ~ del 2~ ~ del 3~ ~ del 3~ ~ del 4~ ~ del 4~	void loop() { //exécutées en boucle digitalWrite(led1, HIGH); //del 1 delay(1000); //del 1 digitalWrite(led1, LOW); //del 1 //del 2 //del 2 //del 2 //del 3 //del 3 //del 3 //del 4 //del 4 //del 4 }
Fin	

A l'aide de l'algorithme, compléter le programme en C++ de l'arduino.
Le faire vérifier par le professeur.

Tester le programme dans arduino.

Votre programme répond-il au cahier des charges :

V. 2^{ème} Algorithme : structures conditionnelles

Un client souhaite superviser la tension aux bornes d'un potentiomètre à l'aide de 2 DEL.
 Si la tension est inférieure à 2,5V DEL1 allumée.
 Sinon DEL2 allumée.

Remarque ; Arduino dispose en interne d'un module de « conversion analogique-numérique » qui permet d'obtenir une valeur entre 0 et 1023 correspondant au niveau de la tension entre 0 et 5V présente sur la broche. A0 à A5.

Une partie de l'algorithme est donné ci-dessous. Compléter le, pour répondre au cahier des charges.

Algorithme	Programme en C++
Constante led1 : entier =2 ~ déclaration de la del 1 ~ ~ déclaration de la del 2 ~	const int led1 = 2; //déclaration de la del 1
Variable mesure : entier ~ récupéra la mesure ~	int mesure ; //récupéra la mesure
Procédure setup() Début ~ Configuration initiale ~ initialiser led1 comme une sortie ~ del 1~ ~ del 2~	void setup() { //Configuration initiale: pinMode(led1, OUTPUT); //del 1 //del 2 }
Fin	
Procédure loop() Début ~ exécutées en boucle ~ mesure = valeur de A0 éteindre led1 ~ del 1~ éteindre led2 ~ del 2~ ~ structure conditionnelle ~ Si mesure < 512 Alors led1 allumée ~ del 1~ Sinon led2 allumée ~ del 2~ FinSi	void loop() { //exécutées en boucle mesure = analogRead(A0); //del 1 //del 2 ~ structure conditionnelle ~ if (mesure < 512) //del 1 else //del 2 }
Fin	

A l'aide de l'algorithme, compléter le programme en C++ pour l'arduino.
 Le faire vérifier par le professeur.

Tester le programme dans arduino.

Votre programme répond-il au cahier des charges :

VI. 3^{ème} Algorithme : structures conditionnelles imbriquées

Le client souhaite superviser la tension aux bornes du potentiomètre à l'aide de 4 DEL.

Si la tension est inférieure à 1,25V DEL1 allumée.

Si la tension est inférieure à 2,5V DEL2 allumée.

Si la tension est inférieure à 3,75V DEL1 allumée.

Sinon DEL4 allumée.

Ecrire l'algorithme pour répondre au cahier des charges.

Algorithme	Programme en C++

A l'aide de l'algorithme, écrire le programme en C++.

Tester le programme dans arduino.

Le faire vérifier par le professeur.

Votre programme répond-il au cahier des charges?

VII. Conclusion