

Déboguer (dépanner) un programme (système) :

Les meilleures pratiques de codage pour éviter d'avoir des problèmes

K. Boudjelaba

k.boudjelaba@carnus.fr

5 janvier 2023

1 Introduction

”Tout le monde sait que le débogage est deux fois plus difficile que l’écriture d’un programme. Donc, si vous êtes aussi intelligent que vous pouvez l’être quand vous écrivez votre code, comment allez-vous déboguer ?”

- Nous écrivons tous du code bogué, acceptez-le et faites avec.
- Écrivez votre code avec le test et le débogage à l’esprit.
- Keep It Simple, Stupid (KISS) (gardez-le simple, stupide).
 - Quelle est la chose la plus simple qui pourrait fonctionner ?
- Ne vous répétez pas.
 - Chaque bout de connaissance doit avoir une seule représentation autorisée et non ambiguë dans un système.
 - Constantes, algorithmes, etc.
- Essayez de limiter l’interdépendance de votre code. (Loose Coupling)
- Donnez à vos variables, fonctions et modules des noms explicites (pas des noms mathématiques).

2 Le travail de débogage

Si vous avez un bogue non banal, c’est là que les stratégies de débogage vont rentrer en ligne de compte. Il n’y a pas de solution miracle, les stratégies aideront. Pour déboguer un problème donné, la situation favorable est quand le problème est isolé dans un petit nombre de lignes de code, en dehors de framework ou de code d’application, avec un cycle court de modification, lancement, échec.

- Faites échouer le code de façon fiable : trouvez un cas de test qui fait échouer le code à chaque fois.
- Diviser et conquérir : une fois que vous avez un cas de test échouant, isolez le code coupable.
 - Quel module.
 - Quelle fonction.
 - Quelle ligne de code.
- Isolez une petite erreur reproductible : un cas de test.
- Changez une seule chose à chaque fois et réexécutez le cas de test d’échec.
- Utilisez le débogueur pour comprendre ce qui ne va pas.
- Prenez des notes et soyez patient, ça peut prendre un moment.

Une fois que vous avez procédé à cette étape, isolez un petit bout de code reproduisant le bogue et corrigez celui-ci en utilisant ce bout de code, ajoutez ce code dans votre suite de test.

3 Mesurer le courant qui traverse une LED

On veut mesurer le courant qui traverse la LED à l'aide de la carte **Arduino UNO**.
Le programme pour effectuer cette mesure est donné ci-dessous :

Programme Arduino : Mesurer une tension

BTS SN

```
1 // Entrée analogique utilisée pour mesurer la tension
2 #define entreeAnalogique 2
3
4 void setup()
5 {
6     // Initialisation de la connexion série avec le moniteur
7     Serial.begin(115200) ;
8 }
9
10 void loop()
11 {
12     // Lecture de la valeur analogique à mesurer
13     int valeurLue = analogRead(entreeAnalogique) ;
14     // Conversion de la valeur lue en une tension en centi-Volts
15     float tensionLue = map(valeurLue, 0, 511, 0, 500);
16     // Envoi pour affichage sur le moniteur série de la tension mesurée
17     Serial.print("Tension : ") ;
18     Serial.print(tensionLue / 50.0) ; // Afficher la valeur en Volts
19     Serial.println(" Volts") ;
20     // Attente d'une seconde (1000 ms) entre deux mesures et affichages
21     delay(1000) ;
22 }
```

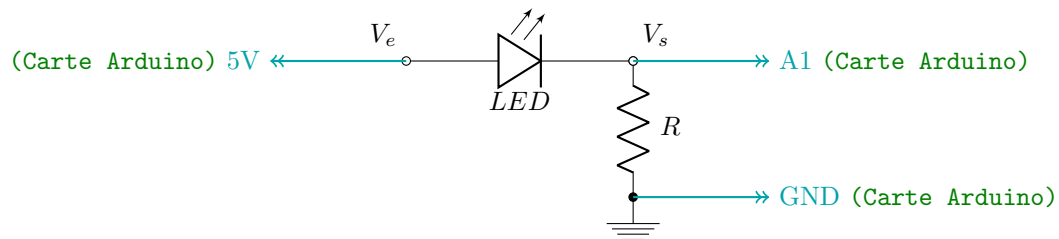


FIGURE 1 – Mesure de courant

Avec $R = 330 \, \Omega$

Vérifier et corriger le programme pour mesurer le courant à travers la LED.

3.1 Programmes Python

a. Soit le programme ci-dessous :

Programme 1 Python	BTS SN
<pre>"""Small snippet to raise an IndexError.""" def index_error(): lst = list('foobar') print(lst[len(lst)]) if __name__ == '__main__': index_error()</pre>	

- Exécuter le programme (ce programme contient des erreurs).
- Peut-on corriger cette erreur ?
- Exécuter l'instruction `%debug` puis taper :
 - `list`
 - `len(lst)`
 - `print(lst[len(lst)-1])`
 - `quit`
- Corriger l'erreur.

b. Soit le programme ci-dessous :

Programme 2 Python	BTS SN
<pre># Programme pour calculer la distance entre deux points import numpy as np import matplotlib.pyplot as plt %matplotlib inline # Coordonnées des points A et B # sous forme A = abscisse de A, ordonnée de A A = 2, 7 B = -1, 3 D = A[1]-A[0]**2+(B[1]-B[0])**2 print("La distance entre le point A et le point B est :",D)</pre>	

- Le programme contient-il des erreurs ?
- Si oui, corrigez les erreurs.

3.2 Programmes C++

a. Soit le programme ci-dessous :

Ce programme prend un entier positif de l'utilisateur (supposons que l'utilisateur a entré n) puis, ce programme affiche la valeur de la somme $1 + 2 + 3 + \dots + n$ et la moyenne de la séquence des nombres $(1, 2, \dots, n)$.

Programme 1 C++ :

BTS SN

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int n, sum = 0;
7
8     cout << "Entrer un entier positif : ";
9
10    for (int i = 1; i <= n; ++i) {
11        sum *= i;
12    }
13
14    cout << "Somme = " << sum;
15    cout << "Moyenne = " << sum/2;
16    return 0;
17 }
```

— Corriger les erreurs de ce programme.