

# Free Basic to C compiler

Jérémy Bardon

15 mars 2015

## 1 Introduction

Le but de ce projet est de créer un compilateur capable de convertir un code écrit en Free Basic dans le langage C.



FIGURE 1 – Rôle du compilateur

Le programme en C ainsi généré sera compilable avec *gcc* et effectuera les mêmes opérations que le programme original.

## 2 Adaptations

Par défaut, un programme écrit en free basic dispose comme en C d'un certain nombre de fonctions que l'on peut utiliser sans inclure des programmes externes. Cependant, ces fonctions « basiques » ne sont pas équivalentes dans les deux langages.

L'exemple le plus simple est la fonction qui permet d'afficher un message : en free basic il s'agit de la fonction *print* qui est incluse par défaut mais en C il est nécessaire d'inclure *stdio.h*.

Une autre différence de taille est la présence – en C – d'une routine principale (*main*) qui doit être présente dans tout programme écrit en C. Ce n'est pas le cas du free basic qui propose une structure plus libre du programme.

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char** argv){

    // Converted free basic code
    // comes here

    return EXIT_SUCCESS;
}
```

Listing 1 – Programme C englobant

Ce sont ces différences qui amènent à devoir proposer une structure englobante d'un programme dans laquelle on va ensuite insérer le code free basic converti.

### 3 Fonctionnalités implémentées

- Fonction à 1 argument
- Déclaration de constantes (entier et chaîne de caractères)
- Déclaration et changement de valeur de variables (entier et chaîne de caractères)
- Commentaire mono-ligne
- Condition simple (pas de close sinon)
- Report du numéro de ligne et de caractère en cas d'erreur lors du parsing

### 4 Grammaire

### 5 Reconnaissance des erreurs