Exo1 :

1.feuille

2. a. oui

b. non car que les enfants qui prenne le type des parents pas l’inverse

c. oui car BlocF fils de Bloc

d. non car il est maintenant caster en BlocF (BlocF et BlocR ne communique pas entre eux)

3. on utilise des stacks de bloc car on utilise des fonctions avec différents retour et arguments

4. le code appelant : ce qui gère l’appel et le retour

Le code appelé : ce qui exécute la fonction (donc on met tout ce qui se passe dans la fonction est dans le code appelé)

Les constructeurs : le code appelant

Bloc :

getAdresse() : le code appelé //get adresse c la valeur de ic ou on doit retourner apres la fin de la fonction //car on en a besoin pour sortir du code

BlocR :

getVal() : dans le code appelant //car c’est la valeur a retourner dans le code appelé

setVal() : dans le code appelé //car la valeur de retour est calculé a l’execution de la fonction

BlocF :

getArg1() : code appelant //car c’est les argments a mettre en parametre de la fonction

getVar1() : code appelé //variable utiliser que dans la fonction reste dedans

setVar1() : code appelé

BlocG :

getArg1() : code appelant

getArg2() : code appelant

Exo2 :

1. le programme affiche 42

2.