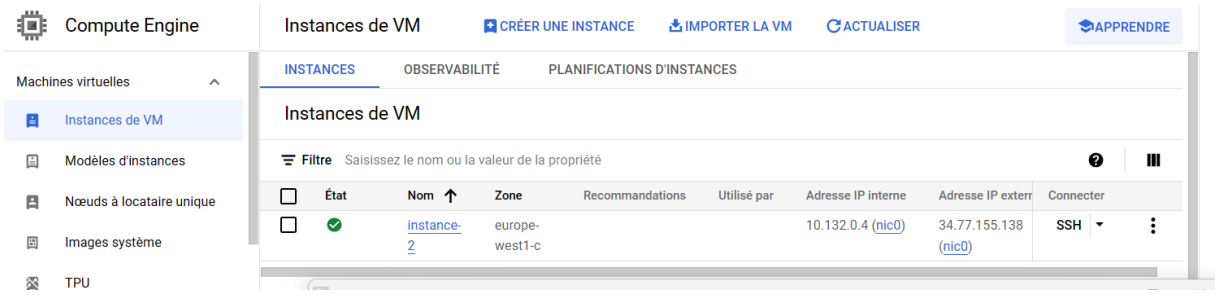
Rapport Programmation par composant

Dans le cadre du cours de programmation par composant, nous avons instancié une machine virtuelle sur google cloud pour pouvoir exécuter les différents programmes.



Nous avons clôné le répertoire git et installé les commandes nécessaires pour la compilation :

```
anamshaikhyunus@instance-2:~$ git clone https://github.com/AnamNasar/lib_partageable_2023.git
Cloning into 'lib_partageable_2023'...
remote: Enumerating objects: 37, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 37 (delta 0), reused 0 (delta 0), pack-reused 36
Receiving objects: 100% (37/37), 7.07 KiB | 7.07 MiB/s, done.
Resolving deltas: 100% (9/9), done.
```

1. Compilation et exécution

La commande make pour la compilation des fichiers sources :

```
anamshaikhyunus@instance-2:~/lib_partageable_2023$ make
cd main && make install
make[1]: Entering directory '/home/anamshaikhyunus/lib_partageable_2023/main'
mkdir ../bin
for component in Composant1 Composant2; do \
        cd ../$component; \
        make install;  \
done
make[2]: Entering directory '/home/anamshaikhyunus/lib_partageable_2023/Composant1'
mkdir -p ../lib
g++ -fPIC -I../interfaces -c Composant1.cpp -o Composant1.o
gcc -o libComposant1.so -shared  Composant1.o
cp libComposant1.so ../lib/libComposant1.so
make[2]: Leaving directory '/home/anamshaikhyunus/lib_partageable_2023/Composant1'
make[2]: Entering directory '/home/anamshaikhyunus/lib_partageable_2023/Composant2'
g++ -fPIC -I../interfaces -c Composant2.cpp -o Composant2.o
gcc -o libComposant2.so -shared  Composant2.o
cp libComposant2.so ../lib/libComposant2.so
make[2]: Leaving directory '/home/anamshaikhyunus/lib_partageable_2023/Composant2'
g++ -fPIC -I../interfaces -c main.cpp -o main.o
g++ main.o -L ../lib -lComposant1 -lComposant2 -o main
cp main ../bin; \
cp ../lib/libComposant1.so ../lib/libComposant2.so ../bin
make[1]: Leaving directory '/home/anamshaikhyunus/lib_partageable_2023/main'
cd bin; \
LD_LIBRARY_PATH=. ./main
Composant 1 version 1.2.2.0
valeur 1 :8 valeur 2 :15
anamshaikhyunus@instance-2:~/lib_partageable_2023$
```

2. Objdump

Exercice : objdump -T main

```
anamshaikhyunus@instance-2:~/lib_partageable_2023$ cd main
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ objdump -T main

main:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4endlIcSt11char_traitsIcEERSt13basic_ostreamI
T_T0_ES6_
0000000000000000      DF *UND*  0000000000000000              _Z10composant2ii
0000000000000000      DF *UND*  0000000000000000              _Z10composant1ii
0000000000000000      DF *UND*  0000000000000000              _Z20getComposant1Versionv
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_atexit
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_E
S5_PKc
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEPFRSoS_E
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4cout
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitC1Ev
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEi
0000000000000000  w   D  *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __libc_start_main
0000000000000000  w   D  *UND*  0000000000000000              __gmon_start__
0000000000000000  w   D  *UND*  0000000000000000              _ITM_registerTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitD1Ev
0000000000000000  w   DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_finalize
```

Exercice : objdump -T libComposant1.so

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ cd ..
anamshaikhyunus@instance-2:~/lib_partageable_2023$ cd Composant1
anamshaikhyunus@instance-2:~/lib_partageable_2023/Composant1$ objdump -T libComposant1.so

libComposant1.so:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000  w   D  *UND*  0000000000000000 __cxa_finalize
0000000000000000  w   D  *UND*  0000000000000000 _ITM_registerTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 _ITM_deregisterTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 __gmon_start__
0000000000001138 g    DF .text  000000000000000d _Z20getComposant1Versionv
0000000000001124 g    DF .text  0000000000000014 _Z16addition_interneii
0000000000001105 g    DF .text  000000000000001f _Z10composant1ii


anamshaikhyunus@instance-2:~/lib_partageable_2023/Composant1$
```

Exercice : objdump -C -T main

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ objdump -C -T main

main:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::basic_ostream<char, std::char_traits<char> >
& std::endl<char, std::char_traits<char> >(std::basic_ostream<char, std::char_traits<char> >&)
0000000000000000      DF *UND*  0000000000000000              composant2(int, int)
0000000000000000      DF *UND*  0000000000000000              composant1(int, int)
0000000000000000      DF *UND*  0000000000000000              getComposant1Version()
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_atexit
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::basic_ostream<char, std::char_traits<char> >
& std::operator<< <std::char_traits<char> >(std::basic_ostream<char, std::char_traits<char> >&, char const*)
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ostream::operator<<(std::ostream& (*)(std::o
stream&))
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 std::cout
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ios_base::Init::Init()
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ostream::operator<<(int)
0000000000000000  w   D  *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __libc_start_main
0000000000000000  w   D  *UND*  0000000000000000              __gmon_start__
0000000000000000  w   D  *UND*  0000000000000000              _ITM_registerTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ios_base::Init::~Init()
0000000000000000  w   DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_finalize
```

Exercice : objdump -C -T libComposant1

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/Composant1$ objdump -C -T libComposant1.so

libComposant1.so:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000  w   D  *UND*  0000000000000000 __cxa_finalize
0000000000000000  w   D  *UND*  0000000000000000 _ITM_registerTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 _ITM_deregisterTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 __gmon_start__
0000000000001138 g    DF .text  000000000000000d getComposant1Version()
0000000000001124 g    DF .text  0000000000000014 addition_interne(int, int)
0000000000001105 g    DF .text  000000000000001f composant1(int, int)
```

Après avoir décommenté extern C :

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ objdump -T main

main:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4endlIcSt11char_traitsIcEERSt13basic_ostreamI
T_T0_ES6_
0000000000000000      DF *UND*  0000000000000000              _Z10composant2ii
0000000000000000      DF *UND*  0000000000000000              _Z10composant1ii
0000000000000000      DF *UND*  0000000000000000              _Z20getComposant1Versionv
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_atexit
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_E
S5_PKc
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEPFRSoS_E
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4cout
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitC1Ev
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEi
0000000000000000  w   D  *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __libc_start_main
0000000000000000  w   D  *UND*  0000000000000000              __gmon_start__
0000000000000000  w   D  *UND*  0000000000000000              _ITM_registerTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitD1Ev
0000000000000000  w   DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_finalize
```

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ objdump -C -T main

main:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::basic_ostream<char, std::char_traits<char> >
& std::endl<char, std::char_traits<char> >(std::basic_ostream<char, std::char_traits<char> >&)
0000000000000000      DF *UND*  0000000000000000              composant2(int, int)
0000000000000000      DF *UND*  0000000000000000              composant1(int, int)
0000000000000000      DF *UND*  0000000000000000              getComposant1Version()
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_atexit
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::basic_ostream<char, std::char_traits<char> >
& std::operator<< <std::char_traits<char> >(std::basic_ostream<char, std::char_traits<char> >&, char const*)
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ostream::operator<<(std::ostream& (*)(std::o
stream&))
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 std::cout
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ios_base::Init::Init()
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ostream::operator<<(int)
0000000000000000  w   D  *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __libc_start_main
0000000000000000  w   D  *UND*  0000000000000000              __gmon_start__
0000000000000000  w   D  *UND*  0000000000000000              _ITM_registerTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 std::ios_base::Init::~Init()
0000000000000000  w   DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_finalize
```

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/Composant1$ objdump -T libComposant1.so

libComposant1.so:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000  w   D  *UND*  0000000000000000 __cxa_finalize
0000000000000000  w   D  *UND*  0000000000000000 _ITM_registerTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 _ITM_deregisterTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 __gmon_start__
0000000000001138 g    DF .text  000000000000000d _Z20getComposant1Versionv
0000000000001124 g    DF .text  0000000000000014 _Z16addition_interneii
0000000000001105 g    DF .text  000000000000001f _Z10composant1ii
```

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/Composant1$ objdump -C -T libComposant1.so

libComposant1.so:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000  w   D  *UND*  0000000000000000 __cxa_finalize
0000000000000000  w   D  *UND*  0000000000000000 _ITM_registerTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 _ITM_deregisterTMCloneTable
0000000000000000  w   D  *UND*  0000000000000000 __gmon_start__
0000000000001138 g    DF .text  000000000000000d getComposant1Version()
0000000000001124 g    DF .text  0000000000000014 addition_interne(int, int)
0000000000001105 g    DF .text  000000000000001f composant1(int, int)
```

Analyse : En ajoutant extern "C" autour de la déclaration de la fonction composant2, nous indiquons au compilateur C++ de ne pas effectuer de "mangling" sur ce symbole, de sorte que le nom réel de la fonction exposée reste inchangé et peut être utilisé dans du code C.

3. dlopen et dlsym

ajout de -ldl dans Makefile :

```
  GNU nano 5.4                                          Makefile
CPPFLAGS=-I../interfaces
BIN_DIR=../bin
COMPONENTS_NEEDED=Composant1 Composant2
COMPONENTS_DIR=../lib
COMPONENTS_FILES=${COMPONENTS_NEEDED:%=${COMPONENTS_DIR}/lib%.so}
LD_COMPONENTS=${COMPONENTS_NEEDED:%=-l%}
LDFLAGS=-L${COMPONENTS_DIR} ${LD_COMPONENTS}
LDLIBS = -ldl
EXE = main
OBJS = main.o

include ../gcc.mk

all: ${COMPONENTS_FILES} ${OBJS}
        ${CXX} ${LDFLAGS} ${OBJS} ${LDLIBS} ${LD_COMPONENTS} -ldl -o ${EXE}

${COMPONENTS_FILES}:
        for component in ${COMPONENTS_NEEDED}; do \
                cd ../$$component; \
                ${MAKE} install;  \
        done

clean:
        rm -f ${OBJS} ${EXE} ${COMPONENTS_FILES}; \
        for component in ${COMPONENTS_NEEDED}; do \
                cd ../$$component; \
                ${MAKE} clean;  \
        done; \
        rm -fr ${COMPONENTS_DIR}
        rm -fr ${BIN_DIR}

install: ${BIN_DIR} all
        cp ${EXE} ${BIN_DIR}; \
        cp ${COMPONENTS_FILES} ${BIN_DIR}

${BIN_DIR}:
        mkdir $@
```

main.cpp pour illustrer les utilisations de dlopen et dlsym :

```
  GNU nano 5.4                                        main.cpp
#include <iostream>
#include <dlfcn.h>

typedef int (*Composant1Func)(int, int);
typedef int (*Composant2Func)(int, int);

int main(int argc, char** argv) {
    int data1 = 3;
    int data2 = 5;
    int valeur1;
    int valeur2;

    void* handle1;
    void* handle2;
    Composant1Func composant1;
    Composant2Func composant2;

    // Ouvrir la bibliothèque libComposant1.so
    handle1 = dlopen("/home/anamshaikhyunus/lib_partageable_2023/lib/libComposant1.so", RTLD_LAZY);
    if (!handle1) {
        std::cerr << "Erreur lors de l'ouverture de libComposant1.so : " << dlerror() << std::endl;
        return 1;
    }

    // Résoudre le symbole 'composant1'
    composant1 = (Composant1Func)dlsym(handle1, "composant1");
    if (!composant1) {
        std::cerr << "Erreur lors de la résolution du symbole composant1 : " << dlerror() << std::endl;
        dlclose(handle1);
        return 1;
    }

    // Ouvrir la bibliothèque libComposant2.so
    handle2 = dlopen("/home/anamshaikhyunus/lib_partageable_2023/lib/libComposant2.so", RTLD_LAZY);
    if (!handle2) {
        std::cerr << "Erreur lors de l'ouverture de libComposant2.so : " << dlerror() << std::endl;
        dlclose(handle1);
```

Objdump :

```
anamshaikhyunus@instance-2:~/lib_partageable_2023/main$ objdump -T main

main:     file format elf64-x86-64

DYNAMIC SYMBOL TABLE:
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4endlIcSt11char_traitsIcEERSt13basic_ostreamI
T_T0_ES6_
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 dlerror
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 dlclose
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_atexit
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZStlsISt11char_traitsIcEERSt13basic_ostreamIcT_E
S5_PKc
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEPFRSoS_E
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4cout
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 dlopen
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitC1Ev
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSolsEi
0000000000000000  w   D  *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 dlsym
0000000000000000      DF *UND*  0000000000000000  GLIBC_2.2.5 __libc_start_main
0000000000000000  w   D  *UND*  0000000000000000              __gmon_start__
0000000000000000  w   D  *UND*  0000000000000000              _ITM_registerTMCloneTable
0000000000000000      DO *UND*  0000000000000000  GLIBCXX_3.4 _ZSt4cerr
0000000000000000      DF *UND*  0000000000000000  GLIBCXX_3.4 _ZNSt8ios_base4InitD1Ev
0000000000000000  w   DF *UND*  0000000000000000  GLIBC_2.2.5 __cxa_finalize
```