

**TD10:**

```

user@PNS-VirtualBox:~/Documents/ProgSys/td10$ java -Djava.library.path=./lib exo1
9466
9466
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ java -Djava.library.path=./lib exo2
9486
9486
9486
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ java -Djava.library.path=./lib exo3
9504
9504
9504
Le père
Le fils
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ cp exo3 java exo4 java

```

```

user@PNS-VirtualBox:~/Documents/ProgSys/td10$ ./exo5.exe
./exo5.exe: error while loading shared libraries: libjvm.so: cannot open shared object file: No such file or directory
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ ldd exo5.exe
        linux-vdso.so.1 (0x00007fff16eef000)
        libjvm.so => not found
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f3488615000)
        /lib64/ld-linux-x86-64.so.2 (0x00007f348881f000)
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ locate libjvm.so
/home/user/.vscode/extensions/redhat.java-1.5.0-linux-x64/jre/17.0.2-linux-x86_64/lib/server/libjvm.so
/usr/lib/jvm/java-11-openjdk-amd64/lib/server/libjvm.so
/usr/share/gdb/auto-load/usr/lib/jvm/java-11-openjdk-amd64/jre/lib/server/libjvm.so-gdb.py
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ export LD_LIBRARY_PATH=/usr/lib/jvm/java-11-openjdk-amd64/lib/server/
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ ./exo5.exe
Impossible de trouver la classe exo1
user@PNS-VirtualBox:~/Documents/ProgSys/td10$ ldd exo5.exe
        linux-vdso.so.1 (0x00007ffec67d2000)
        libjvm.so => /usr/lib/jvm/java-11-openjdk-amd64/lib/server/libjvm.so (0x00007fclcf458000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fclcf255000)
        libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007fclcf24f000)
        libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007fclcf22c000)
        libstdc++.so.6 => /lib/x86_64-linux-gnu/libstdc++.so.6 (0x00007fclcf04a000)
        libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007fclcee0b000)
        libgcc_s.so.1 => /lib/x86_64-linux-gnu/libgcc_s.so.1 (0x00007fclcee0e000)
        /lib64/ld-linux-x86-64.so.2 (0x00007fclcf1d0718000)
user@PNS-VirtualBox:~/Documents/ProgSys/td10$

```

**Exercice 6**

La première méthode consiste à créer une JVM en C++ et à exécuter le programme Java de l'exercice 1 dedans.

Une fois l'exécution terminée, le programme continue son exécution, par exemple si on met des printf ils s'exécuteront.

En revanche, avec une exécution de execXX pour exécuter le programme Java, le programme java se lance et toutes les lignes écrites après ne seront pas exécutées, sauf si nous faisons un fork (dans ce cas l'un des 2 continuera).

**Exercice 7**

Il est possible de surcharger une méthode du code Java, nous avons la possibilité de surcharger par exemple System.out.println afin de faire en sorte que cette méthode ne puisse plus rien afficher en sortie.

**Exercice 8**

8. Une idée pourrait être d'ajouter une clé en attribut de la classe.

Ensuite, avant chaque exécution de méthode, on pourrait vérifier si elle est identique à celle du programme que l'on souhaite et ainsi empêcher les autres de se lancer.