Project 1 : Warmup (LLVM)

(Submitted by Gowtham Tumati, as a part of CS201 for Winter 2020)

Step 1 :

Successful installation of the LLVM environment

Step 2 :

Learning the usage of basic commands and translating between different code representations.

Translations between code representations:

i) source (.c) to binary (executable)

```
clang test.c (an executable file named ‘a’ is generated in the list of files)
```

ii) source (.c) to objective (.o)

```
clang -c test.c
```

iii) source (.c) to machine assembly (.s)

```
clang -c test.c -emit-llvm -o assemblysamp.s
```

iv) source (.c) to LLVM bitcode (.bc)

```
clang -c test.c -emit-llvm -o bitcodesamp.bc
```

v) source (.c) to LLVM IR (.ll)

```
clang -emit-llvm -S test.c -o ircode.ll
```

vi) LLVM IR (.ll) to LLVM bitcode (.bc)

```
llvm-as ircode.ll -o bcsampl.bc
```

vii) LLVM bitcode (.bc) to LLVM IR (.ll)

```
llc bitcodesamp.bc -o irconv.ll
```

viii) LLVM IR (.ll) to machine assembly (.s)

```
llc ircode.ll -o massembly.s
```

ix) interpret the LLVM IR (which directly prints the output without compilation)

```
lli bitcodesamp.bc
```

By doing the translation of the code representations, I have made the use of clang, llvm, llvm-as, llc, lli and other commands that were specified in the original document.

Generation of CFG for the test file has been done with the use of the command mentioned below :

```
opt -dot-cfg bitcodesamp.bc
```

Step 3 :

This is the report that consists of the list of experiments performed and were successful. Also, the files which are used as a part of this warmup were also included in the submission.

-THE END-