# 1 Loops: 'while', 'for' and 'do while'

```
a) high = n - 1;
   for(low = 0; low <= high; ++low){
       //do  some actions
   }
```

```
b) low = 0
   high = n - 1;
   if(n <= 0){
       return;
   } else {
       do{
           //do some actions
           ++low;
       }while{low <= high}
   }
```

# 2 goto & label, switch, break and continue

```
a) i = 0;
   LOOP: if(i < n){
       i++;
       //do some actions
       goto LOOP;
   }
```

```
b) if(i == 1){
       printf("case 1\n");
       goto end;
   } else if (i == 2){
       printf("case 2\n");
   }
   printf("default case\n");
   end:
```

c)
```c
for (i = 0; i < n; i++) {
  // do some actions 1
  if (<cond1>) goto stop;
  // do some actions 2
}
stop:
```

d)
```c
for (i = 0; i < n; i++) {
  // do some actions 1
  if (<cond1>) goto next;
  // do some actions 2
  next:
}
```

# 3   gdb - 1$^{st}$ visit

```
$ (gdb) b 4
$ (gdb) b 7
$ (gdb) r
$ (gdb) c
$ (gdb) w i
$ (gdb) w res
$ (gdb) c
$ (gdb) c
$ (gdb) c
$ (gdb) c
$ (gdb) c
$ (gdb) c
```

# 4   gdb - 2$^{nd}$ visit

Coudn't generate the core file with the Windows Subsystems for Linux.

# 5 Abstact Syntax Tree (AST) and Control Stack (CS) for an expression

```
              ;
              |
              =
            /  \
         &n     +
               /  \
             $n     *
                   /  \
                 $m    2


              |
             *=
           / | \
        &n  $n  +
               /  \
              1    $m
```