

# Code Generation 2

Prat Gary

November 14, 2019

## 1 Exercise 79

```
1.
for (var = lbound; var < ubound;step) {
    S(var)
}
```

```
GCStm(for(v=lb;v < ub;step){S(v)})=
let end,debut := new Label()
(Clb,Rv) ← GCAexp(lb)
(Cub,Ri1) ← GCAexp(ub)
(Cst,Rs)← GCAexp(step)
Clb||
debut:||
Cub||
Cst||
CMP Rv Ri1||
BGE end||
GCStm(S(v),Rv)||
ADD Rv,Rv,Rs||
BA debut||
end:
```

```
2.
Prologue(44)
```

```
SUB R0,R0,R0
ADD R1,R0,R0
ST R1,[FP-44]
debut:
ADD R2,R0,10
LD R1,[FP-44]
CMP R1,R2
BGE end:
ADD R3,R0,R0
LD R1,[FP-44]
ADD R4,R1,1
MUL R5,R4,4
```

```

ST R3,[FP-R5]
LD R1,[FP-44]
ADD R1,R1,1
ST R1,[FP-44]
BA debut
end:
Epilogue

```

3. Non fait, trop similaire.

## 2 Exercise 80

1. repeat S until b

```

GCStm(repeat S until b)=
let begin,end:=new Label()
begin:||
GCStm(S)||
GCBexp(b,end,begin)||
end:

```

2. b1 xor b2

```

GCBexp(b1 xor b2,l1,l2)=\\ l1,l2 =Label
let bt1,bf1= new label()
GCBexp(b1,bt1,bf1)
bt1:
GCBexp(b2,l2,l1)\\ b1=True
bf1:
GCBexp(b2,l1,l2)\\ b1=False

```

3. b ? e1 : e2

```

\\ Pas forcément juste, en attente de correction.
BCAexp(b ? e1 : e2)=
let R1 =new Alloc()
let b1,b2 =new Label()
GCBexp(b,b1,b2)
b1:
ADD R1,e1,0
b2:
ADD R1,e2,0

```