

Corrigé de l'exercice 1 de l'interrogation d'architecture

```
Int a,b,s,i ;
i=0;
s=0;
If(a<0) a=-a;
While (i<100)
{ s=s+a-i;
i=i+b;
}
```

Traduction en Mexs:

```
@a=1,@b=2, @s=3,@i=4
fonctions des registres:
R0 contient 0
R1 contient @a, R11 contient a
R2 contient @b, R12 contient b
R3 contient @s, R13 contient s
R4 contient @i, R14 contient i
R15 contient 100
/* initialisation des registres constantes ou adresse
    SETHI R0 0
    SETHI R1 1
    SETHI R2 2
    SETHI R3 3
    SETHI R4 4
    SETHI R15 100
    LD R12 (R0+R2)
/* i=0
    ADD R14 R0 R0
    ST (R0+R4), R14
/*s=0
    ADD R13 R0 R0
    ST (R0+R3), R13
/* traduction du if
    LD R11, (R0+R1)
    ADDcc R11, R0,R11
    BPZ bcl /* branchement si a >=0
    SUB R11,R0,R11
    ST (R0+R1), R11
/* traduction du while
bcl  LD R14, (R0+R4)
    SUBcc R5 R14 R15 /* i-100
    BPZ fin
    SUB R5 R11 R14
    ADD R13 R13 R5
    ST (R0+R3), R13
    ADD R14 R14 R12
    ST (R0+R4), R14
    JMPL bcl
Fin  NOP
```

Traduction en IJVM:

Les variables a,b,s,i ont pour numéro 1,2,3,4. Toute la difficulté vient du fait qu'il n'y a que très peu de branchements conditionnels possibles. On montre ici une solution possible, mais il y en a d'autres.

```
/* i=0
    BIPUSH 0
    DUP
    ISTORE 4
/*s=0
    ISTORE 3
/* traduction du if
    ILOAD 1
    BN neg
    GOTO bcl
neg  BIPUSH 0
    ILOAD 1
    ISUB
    ISTORE 1
/* traduction du while
bcl  ILOAD 4
    BIPUSH 100
    ISUB
    BN bcl2
    GOTO fin
bcl2 ILOAD 1
    ILOAD 4
    ISUB
    ILOAD 3
    IADD
    ISTORE 3
    ILOAD 2
    ILOAD 4
    IADD
    ISTORE 4
    GOTO bcl
Fin  NOP
```