

Projet Systèmes Informatiques

Du compilateur vers le microprocesseur



Thomas **Ballotin**, Terani **Luque**

Développement d'un compilateur

Table des symboles - Programmes C

TDS.h

```
extern int indice;  
void ajouterSymbole(char * nom,int init);  
int indexSymbole(char * nom);  
int initialSymbole(char * nom);  
void initialiserSymbole(char * nom);
```

Développement d'un compilateur

YACC - *If* et *While*

If :

```
t_IF t_PO Condition t_PF {compteurInstruction++;
                          $1=ftell(fichierASM);
                          printf("JMF %d %d\n", $3, $1);
                          fprintf(fichierASM, "JMF %d %d", $3, $1);}
t_AO BodyIF t_AF         {fseek(fichierASM, $1, SEEK_SET);
                          fprintf(fichierASM, "JMF %d %d\n", $3, compteurInstruction+1);
                          fseek(fichierASM, 0, SEEK_END);}
SuiteIf                  {printf("J'ai un if\n");}
;
```

SuiteIf:

While :

```
t_WHILE t_PO Condition t_PF {compteurDebutWhile=compteurInstruction;
                             compteurInstruction++;
                             $1=ftell(fichierASM);
                             printf("JMF %d %d\n", $3, $1);
                             fprintf(fichierASM, "JMF %d %d", $3, $1);}
t_AO BodyIF                 {fprintf(fichierASM, "JMP %d\n", compteurDebutWhile);}
t_AF                       {fseek(fichierASM, $1, SEEK_SET);
                             fprintf(fichierASM, "JMF %d %d\n", $3, compteurInstruction+2);
                             fseek(fichierASM, 0, SEEK_END);
                             printf("J'ai un while\n");}
;
```

```
t_ELSE t_AO BodyIF t_AF {printf("J'ai un else\n");}
| t_ELSE If {printf("J'ai un else if\n");}
| {printf("J'ai fini le if\n");}
;
```

BodyIF:

```
Declarations Instructions {printf("J'ai un body\n");}
| Instructions {printf("J'ai un body\n");}
;
```

Développement d'un compilateur

YACC - *If* et *While*

```
main() {
    int a,b=5 ,c=10,d=5;
    int z;

    a = b* 2+c;
    b = (a+5)*2;
    a=b*5;

    if ( 1 < 2 ) {
        int i=0;
        a=5;
    }else {
        a=12;
    }

    while(a>6){
        a=a-1;
    }

    printf(d);
}
```

```
1  AFC 1 5
2  AFC 2 10
3  AFC 3 5
4  AFC 249 2
5  MUL 249 1 249
6  ADD 249 249 2
7  COP 0 249
8  AFC 249 5
9  ADD 249 0 249
10 AFC 248 2
11 MUL 248 249 248
12 COP 1 248
13 AFC 249 5
14 MUL 249 1 249
15 COP 0 249
```

```
16 AFC 249 1
17 AFC 248 2
18 INF 247 249 248
19 JMF 247 23
20 AFC 5 0
21 AFC 246 5
22 COP 0 246
23 AFC 249 12
24 COP 0 249
25 AFC 249 6
26 SUP 248 0 249
27 JMF 248 32
28 AFC 247 1
29 SOU 247 0 247
30 COP 0 247
31 JMP 26
32 PRI 3
```

Conception d'un microprocesseur

Contient des instructions synchrone ou CLK'EVENT

- le compteur
- le processeur
- la mémoire d'instruction
- le banc de registre
- la mémoire de donnée



Problèmes rencontrés

Conclusion

Merci de votre attention !