

Compilation — TD 2

J. Boender & G. Henry

2009–2010

L’objet de ce TD est d’expérimenter avec les appels de fonction en assembleur mips. En particulier, ce TD utilise une convention d’appel simplifiée par rapport à celle vu en cours et celle qui vous sera demandée dans le projet.

1. Appel de fonctions feuilles

Le but de cet exercice est d’implémenter en assembleur le programme suivant :

```
int main()
{
    printf("%d\n", max(3, 5));
}
```

```
int max(int x, int y)
{
    int m;
    if(x >= y)
        m = x;
    else
        m = y;
    return m;
}
```

1.1 Écrivez en assembleur le code de `main` qui effectue la suite d’actions suivante :

1. empile les paramètres effectifs ;
2. appelle la fonction `max` à l’aide de l’instruction `jal` ;
3. dépile les paramètres ;
4. affiche le résultat, qui se trouve dans `$v0`.

1.2

Écrivez la fonction `max` en assembleur. Cette fonction devra effectuer la suite d’actions suivante :

1. lire ses arguments sur la pile ;
2. exécuter le corps de `max` ;
3. écrire le résultat dans le registre `$v0` ;
4. retourner à l’appelant, à l’adresse stockée dans `$ra`.

2. Appel de fonctions générales

Implémentez en assembleur les fonctions suivantes :

```
int pair(int n)
{
    if(n == 0)
        return 1;
    return impair(n - 1);
}
```

```
int impair(int n)
{
    if(n == 0)
        return 0;
    return pair(n - 1);
}
```

Quel(s) registre(s) supplémentaire(s) faudra-t-il sauvegarder ?