

TP3: recherche dichotomique

Avant de commencer le TP, veuillez à copier l'ensemble du répertoire **\L3Info\ILM\TP34** dans un répertoire de votre compte étudiant. Vous travaillerez ensuite exclusivement sur ce répertoire.

Objet du TP

Dans ce TP, vous allez implémenter une fonction récursive (recherche dichotomique).

Démarrage

Comme pour les TP précédents, un squelette de projet est disponible sur **share**.

1. Copiez le répertoire **G:\l3info\ilm\TP3** dans votre espace personnel.
2. Lancez Altera Monitor en exécutant le fichier **G:\l3info\ilm\monitor.bat**.
3. Ouvrez le fichier **TP3\TP3.amp** dans Altera Monitor.
4. Lorsqu'on dialogue vous propose de charger le processeur Nios II sur la carte, vérifiez que celle-ci est branchée et allumée et cliquez sur Oui.

Travail à effectuer

Le but du TP est d'implémenter la fonction dont le prototype est donné ci-dessous

```
int rechercheDicho(int val, int *tab, int debut, int fin);
```

Cette fonction recherche une valeur **val** entre les positions **debut** et **fin** d'un tableau **tab** préalablement trié, et retourne sa position dans le tableau. Si la valeur n'apparaît pas dans le tableau, la fonction retourne -1.

Pour vous permettre de tester votre fonction, un tableau trié de 100 valeurs entre 1 et 1000 est défini dans le segment de données (fichier **tableau.s**). Son adresse est associée au label **tableau**.

Complétez l'implémentation de la fonction **main**, en considérant que les arguments de la fonction sont passés via **r4-r7** et que la valeur de retour est placée dans **r2**.

```
int main() {
    int val, pos;
    while (1) {
        printf("Entrez un nombre: \n");
        val = read_int();
        pos = rechercheDicho(val, tableau, 0, 99);
        if(pos<0)
            printf("Nombre non trouve.\n");
        else
            printf("La position du nombre est: %d\n", pos);
    }
}
```

Implémentation de la fonction

Implémentez la fonction **rechercheDichot** (...) définie comme suit :

```
int rechercheDicho (int val, int *tab, int debut, int fin) {
    int pos;
    if (debut > fin)
        return -1;
    pos = debut + (fin-debut) / 2;
    if (tab[pos] == val)
        return pos;
    if (tab[pos] < val)
        return rechercheDicho(val, tab, pos+1, fin);
    // tab[pos] > val
    return rechercheDicho(val, tab, debut, pos-1);
}
```

Notes importantes

Dans votre implémentation, vous veillerez tout particulièrement à la gestion de la pile. Les registres utilisés seront sauvegardés selon les conventions vues en cours :

- **r2-r7**: utilisation libre.
- **r8-r15**: si ces registres sont utilisés dans le code appelant, ils doivent être sauvegardés sur la pile avant l'appel.
- **r16-r23**: si ces registres sont utilisés par la fonction appelée, ils doivent être sauvegardés par celle-ci et leur valeur restaurée en fin de fonction.

Compte-rendu

Avant la semaine prochaine, même jour, même heure, envoyez votre code (commenté !) et votre compte-rendu. Ce dernier comportera tous les éléments nous permettant d'apprécier votre travail :

- L'état de votre programme (fonctionnel ou non) ;
- En cas de difficultés, une description de celles-ci et de vos pistes de résolution ;
- Une justification de vos choix d'implémentation.