

Annexe TD n° 4 : Fonctions et procédures

Code généré par le compilateur gcc pour NIOS II.

```

max:
00000258 2005883a    add    r2, r4, zero
0000025c 29000116    blt    r5, r4, +0x4 (00000264)
                else return y;
00000260 2805883a    add    r2, r5, zero
00000264 f800283a    ret
                } 264:    f800283a    ret

main:
00000268 00c003c4    addi   r3, zero, 0xf
0000026c 00800284    addi   r2, zero, 0xa
00000270 deffff04    addi   sp, sp, -0x4
00000274 1009883a    add    r4, r2, zero
00000278 180b883a    add    r5, r3, zero
0000027c dfc00015    stw    ra, 0(sp)
00000280 d0e00715    stw    r3, -32740(gp)
00000284 d0a00615    stw    r2, -32744(gp)
00000288 00002580    call   0x00000096 (00000258, max)
0000028c d0a00515    stw    r2, -32748(gp)
00000290 0005883a    add    r2, zero, zero
00000294 dfc00017    ldw    ra, 0(sp)
00000298 dec00104    addi   sp, sp, 0x4
0000029c f800283a    ret

```

Figure 1 : fonction max et programme principal (exercice 1)

```

change (char v[], int k, int m)
{
char temp;
temp=v[k];
v[k]=v[m];
change:
00000230 218d883a    add    r6, r4, r6
00000234 30c00003    ldbu   r3, 0(r6)
00000238 2149883a    add    r4, r4, r5
0000023c 21400003    ldbu   r5, 0(r4)
00000240 20c00005    stb    r3, 0(r4)
                v[m]=temp;
00000244 31400005    stb    r5, 0(r6)
                }
00000248 f800283a    ret

```

Figure 2 : fonction change pour le tri (exercice 2)

Address	Hex	Assembly	Comment
		tri (char v[], int n)	
		{	
		tri:	
0000024c	defffa04	addi sp, sp, -0x18	
00000250	dc800215	stw r18, 8(sp)	
		int i, j;	
		for (i=n-1;i>0; i--){	
00000254	2cbfffc4	addi r18, r5, -0x1	
00000258	dc400315	stw r17, 12(sp)	
0000025c	dfc00515	stw ra, 20(sp)	
00000260	dc000415	stw r16, 16(sp)	
00000264	dcc00115	stw r19, 4(sp)	
00000268	dd000015	stw r20, 0(sp)	
0000026c	2023883a	add r17, r4, zero	
00000270	04800f0e	bge zero, r18, +0x3c (000002b0)	
		for (j=i-1; j>=0 ; j--){	
00000274	953fffc4	addi r20, r18, -0x1	
00000278	a021883a	add r16, r20, zero	
0000027c	a0000a16	blt r20, zero, +0x28 (000002a8)	
00000280	8ca7883a	add r19, r17, r18	
		if (v[j] >v [i])	change (v,j, i);
00000284	8c05883a	add r2, r17, r16	
00000288	11c00007	ldb r7, 0(r2)	
0000028c	98c00007	ldb r3, 0(r19)	
00000290	800b883a	add r5, r16, zero	
00000294	8809883a	add r4, r17, zero	
00000298	843fffc4	addi r16, r16, -0x1	
0000029c	900d883a	add r6, r18, zero	
000002a0	19c00b16	blt r3, r7, +0x2c (000002d0)	
000002a4	803ff70e	bge r16, zero, -0x24 (00000284)	
000002a8	a025883a	add r18, r20, zero	
000002ac	053ff116	blt zero, r20, -0x3c (00000274)	
		}	
		}	
		}	
000002b0	dfc00517	ldw ra, 20(sp)	
000002b4	dc000417	ldw r16, 16(sp)	
000002b8	dc400317	ldw r17, 12(sp)	
000002bc	dc800217	ldw r18, 8(sp)	
000002c0	dcc00117	ldw r19, 4(sp)	
000002c4	dd000017	ldw r20, 0(sp)	
000002c8	dec00604	addi sp, sp, 0x18	
000002cc	f800283a	ret	
000002d0	00002300	call 0x0000008c (00000230, change)	
000002d4	803feb0e	bge r16, zero, -0x54 (00000284)	
000002d8	003ff306	br -0x34 (000002a8)	

Figure 3 : fonction tri exercice 2

```

main:
000002dc  deffff04      addi    sp, sp, -0x4
000002e0  01800034      orhi    r6, zero, 0x0
000002e4  3181b704      addi    r6, r6, 0x6dc
000002e8  dfc00015      stw     ra, 0(sp)
000002ec  03801904      addi    r14, zero, 0x64
000002f0  03401404      addi    r13, zero, 0x50
000002f4  03001184      addi    r12, zero, 0x46
000002f8  02c00f04      addi    r11, zero, 0x3c
000002fc  02800dc4      addi    r10, zero, 0x37
00000300  02400a04      addi    r9, zero, 0x28
00000304  020008c4      addi    r8, zero, 0x23
00000308  01c00784      addi    r7, zero, 0x1e
0000030c  00c00284      addi    r3, zero, 0xa
00000310  00800644      addi    r2, zero, 0x19
00000314  3009883a      add     r4, r6, zero
00000318  01400284      addi    r5, zero, 0xa
0000031c  33800005      stb     r14, 0(r6)
00000320  33400045      stb     r13, 1(r6)
00000324  33000085      stb     r12, 2(r6)
00000328  32c000c5      stb     r11, 3(r6)
0000032c  32800105      stb     r10, 4(r6)
00000330  32400145      stb     r9, 5(r6)
00000334  32000185      stb     r8, 6(r6)
00000338  31c001c5      stb     r7, 7(r6)
0000033c  30800205      stb     r2, 8(r6)
00000340  30c00245      stb     r3, 9(r6)
00000344  000024c0      call   0x00000093 (0000024c, tri)
00000348  0005883a      add     r2, zero, zero
0000034c  dfc00017      ldw     ra, 0(sp)
00000350  dec00104      addi    sp, sp, 0x4
00000354  f800283a      ret

```

Figure 4 : fonction main pour le tri (exercice 2)