#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <sys/types.h>

#include <sys/wait.h>

**typedef** unsigned char BYTE;

int main(void)

{

pid\_t pid\_son;

BYTE i = 0, j = 0, k = 0, l = 0, m = 0;

int wstatus;

**for**(i = 0; i < 3; i++)

{

**if**((pid\_son = fork()) == 0)

{

printf("|+|I'm the main-process no.1 of main-process no.%d**\n**", i);

**if**(i == 2) *//Begin process level*

{

printf("**\n**");

**for**(j = 0; j < 4; j++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t**|+|+|I'm son no.%d of main-process no.3**\n**", j + 1);

**if**(j == 0) *//Begin sub-process level*

{

printf("**\n**");

**for**(k = 0; k < 3; k++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t\t**|+|+|+|I'm son no.%d of process no.1**\n**", k + 1);

**if**(k == 2)

printf("**\n**"); *//Just to make the printing clearer*

**if**(k == 0) *//Begin subsub-process level*

{

printf("**\n**");

**for**(l = 0; l < 2; l++)

{

**if**((pid\_son = fork()) == 0)

{

pid\_son = getppid();

printf("**\t\t\t**|+|+|+|+|I'm son no.%d: of sub-process no.1: father ID %d**\n**", l + 1, pid\_son);

**if**(l == 1)

printf("**\n**"); *//Just to make the printing clearer*

exit(EXIT\_SUCCESS); *//Kill subsub-process*

}

**else**

{

printf("**\t\t\t**|+|+|+|+|Called subsub-process no.%d: ID %d: from sub-process no.1**\n**", l + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill sub-process*

}

**else**

{

printf("**\t\t**|+|+|+|Called sub-process no.%d: ID %d: from process no.1**\n**", k + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

**if**(j == 1) *//Begin sub-process level*

{

printf("**\n**");

**for**(k = 0; k < 2; k++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t\t**|+|+|+|I'm son no.%d of process no.2**\n**", k + 1);

**if**(k == 1)

printf("**\n**");

**if**(k == 0) *//Begin subsub-process level*

{

printf("**\n**");

**for**(l = 0; l < 2; l++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t\t\t**|+|+|+|+|I'm son no.%d of sub-process no.1**\n**", l + 1);

**if**(l == 1)

printf("**\n**");

**if**(l == 0) *//Begin subsubsub-process level*

{

printf("**\n**");

**for**(m = 0; m < 2; m++)

{

**if**((pid\_son = fork()) == 0)

{

pid\_son = getppid();

printf("**\t\t\t\t**|+|+|+|+|+|I'm son no.%d: of subsub-process no.1: father ID %d**\n**", l + 1, pid\_son);

**if**(m == 1)

printf("**\n**");

exit(EXIT\_SUCCESS); *//Kill subsubsub-process*

}

**else**

{

printf("**\t\t\t\t**|+|+|+|+|+|Called subsubsub-process no.%d: ID %d: from subsub-process no.1**\n**", m + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill subsub-process*

}

**else**

{

printf("**\t\t\t**|+|+|+|+|Called subsub-process no.%d: ID %d: from sub-process no.1**\n**", l + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill sub-process*

}

**else**

{

printf("**\t\t**|+|+|+|Called sub-process no.%d: ID %d: from process no.2**\n**", k + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

**if**(j == 3) *//Begin sub-process level*

{

printf("**\n**");

**for**(k = 0; k < 3; k++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t\t**|+|+|+|I'm son no.%d of process no.4**\n**", k + 1);

**if**(k == 0) *//Begin subsub-process level*

{

printf("**\n**");

**for**(l = 0; l < 2; l++)

{

**if**((pid\_son = fork()) == 0)

{

printf("**\t\t\t**|+|+|+|+|I'm son no.%d of sub-process no.1**\n**", l + 1);

**if**(l == 1) *//Begin subsubsub-process level*

{

printf("**\n**");

**for**(m = 0; m < 2; m++)

{

**if**((pid\_son = fork()) == 0)

{

pid\_son = getppid();

printf("**\t\t\t\t**|+|+|+|+|+|I'm son no.%d: of subsub-process no.2: father ID %d**\n**", l + 1, pid\_son);

**if**(m == 1)

printf("**\n**");

exit(EXIT\_SUCCESS); *//Kill subsubsub-process*

}

**else**

{

printf("**\t\t\t\t**|+|+|+|+|+|Called subsubsub-process no.%d: ID %d: from subsub-process no.2**\n**", m + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill subsub-process*

}

**else**

{

printf("**\t\t\t**|+|+|+|+|Called subsub-process no.%d: ID %d: from sub-process no.1**\n**", l + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill sub-process*

}

**else**

{

printf("**\t\t**|+|+|+|Called sub-process no.%d: ID %d: from process no.4**\n**", k + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

}

exit(EXIT\_SUCCESS); *//Kill process*

}

**else**

{

printf("**\t**|+|+|Called process no.%d: ID %d: from main-process no.3**\n**", j + 1, pid\_son);

waitpid(pid\_son, &wstatus, 0);

}

}

exit(EXIT\_SUCCESS); *//Kill main-process no.3*

}

}

**else**

{

printf("|+|Called main-process no.%d: ID %d: from main-process no.%d**\n**", i + 1, pid\_son, i);

**break**;

}

}

waitpid(pid\_son, &wstatus, 0);

exit(EXIT\_SUCCESS); *//Kill main-process*

**return** 0;}