4 .- Árbol de procesos

el programa genera un arbol de procesos dibujado en el pizarron

Código :

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <sys/types.h>

#include <sys/wait.h>

//void hijos (int vert,int horiz);

int main (void){

int id,j;

pid\_t ID;

id=fork();

if(id==0)

{

for(j=0;j<=9;j++){

id=fork();

if (id==0) {

if(j==0){

for(int k=0;k<=9;k++){

id=fork();wait(0);

if(id==0){

printf("hijo numero 1 %d \n",getpid());

if(k==9){

for( int l=0;l<=9;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==1){

for(int k=0;k<=8;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 2 %d \n",getpid());

if(k==8){

for( int l=0;l<=8;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==2){

for(int k=0;k<=7;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 3 %d \n",getpid());

if(k==7){

for( int l=0;l<=7;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==3){

for(int k=0;k<=6;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 4 %d \n",getpid());

if(k==6){

for( int l=0;l<=6;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==4){

for(int k=0;k<=5;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 5 %d \n",getpid());

if(k==5){

for( int l=0;l<=5;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==5){

for(int k=0;k<=4;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 6 %d \n",getpid());

if(k==4){

for( int l=0;l<=4;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==6){

for(int k=0;k<=3;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 7 %d \n",getpid());

if(k==6){

for( int l=0;l<=3;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==7){

for(int k=0;k<=2;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 8 %d \n",getpid());

if(k==2){

for( int l=0;l<=2;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==8){

for(int k=0;k<=1;k++){

id=fork();

if(id==0){

fflush(stdin);

printf("hijo numero 9 %d \n",getpid());

if(k==8){

for( int l=0;l<=1;l++){

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

}

}

else{break;}

}

}

else if (j==9){

id=fork();

if(id==0){

printf("hijo numero 10 %d \n",getpid());

id=fork();

if(id==0){

printf("hijo del hijo %d \n",getppid());

wait(0);

exit(0);

}

}

else{break;}

}

exit(0);

}

}

}

else {

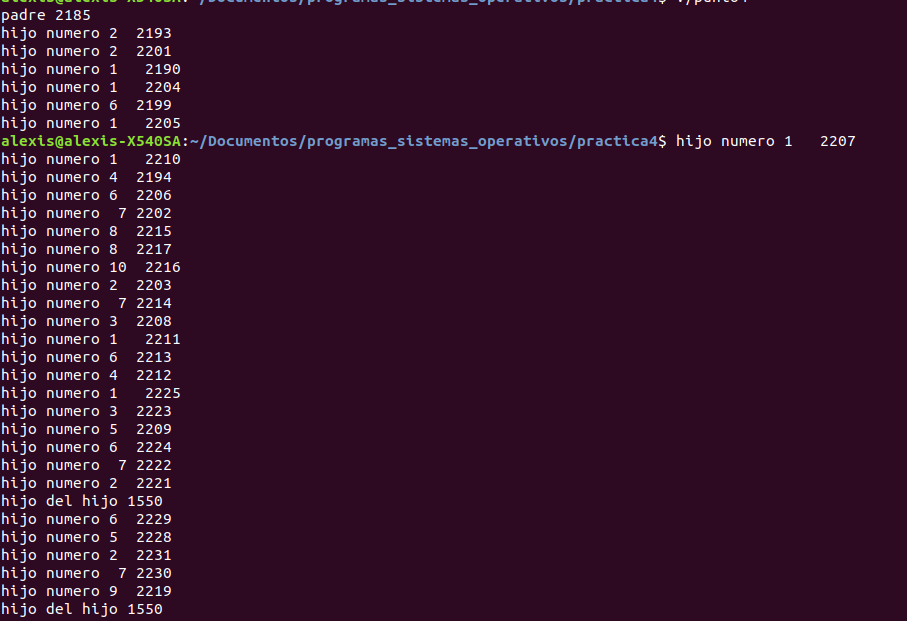
printf("padre %d\n",getpid());

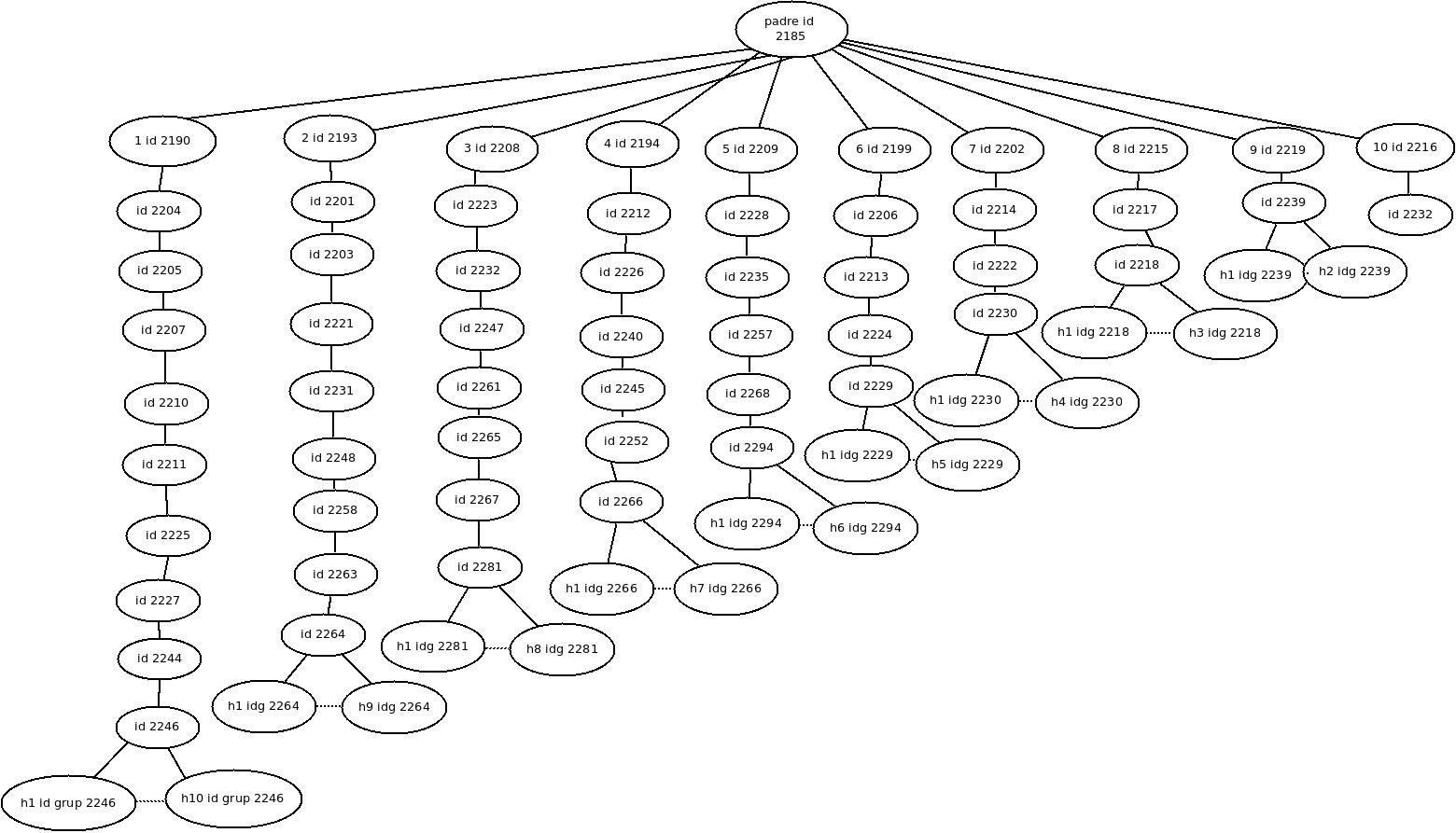
exit(0);

}

}

Salida en la consola :



Dibujo del árbol con los id de procesos generados