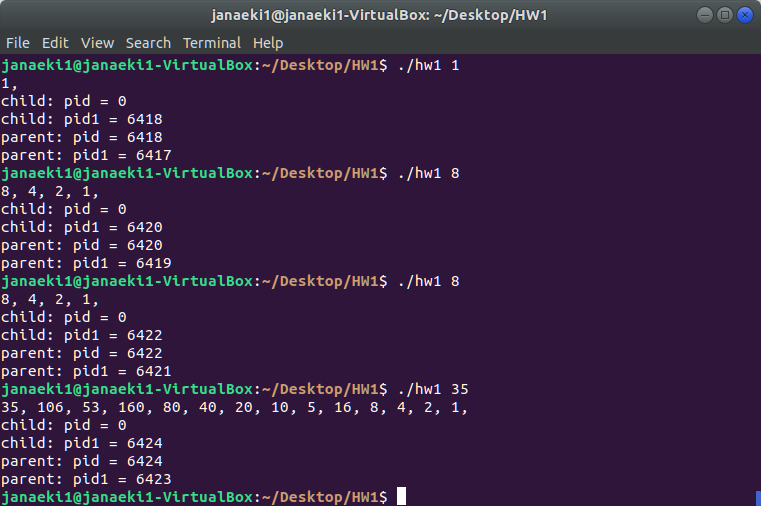
Jerome Anaeki

**Homework 1**

Screenshot

****

Source Code

#include <stdio.h>

#include <unistd.h>

#include <sys/types.h>

/\*

\* hw1.c

\*

\* This program displays the process IDs of both the parent and child processes.

\* It also explores the Collatz conjecture:

\* n = n/2, if n is even

\* OR

\* n = 3 \* n + 1, if n is odd

\* This proves all positive integers will eventually reach 1

\*

\*/

int main(int argc, char \*argv[])

{

int n;

if (argc == 1) {

fprintf(stderr,"Usage: ./hw1 <starting value>\n");

return -1;

}

n = atoi(argv[1]); // n is the input starting value

pid\_t pid, pid1;

/\* fork a child process \*/

pid = fork();

if (pid < 0) { /\* error occurred \*/

fprintf(stderr, "Fork Failed");

return 1;

}

else if (pid == 0) { /\* child process \*/

printf("%d, ", n);

while(n > 1){

if(n%2 == 0){

n /= 2;

}

else{

n=3 \* n + 1;

}

printf("%d, ", n);

}

printf("\n");

pid1 = getpid();

printf("child: pid = %d\n",pid); /\* A \*/

printf("child: pid1 = %d\n",pid1); /\* B \*/

}

else { /\* parent process \*/

wait(NULL);

pid1 = getpid();

printf("parent: pid = %d\n",pid); /\* C \*/

printf("parent: pid1 = %d\n",pid1); /\* D \*/

}

return 0;

}