laborator 10

**Să se implementeze următorul sistem de procese, în care muchiile indică modul de comunicare între procese:**

**1 2**

**\ /**

**3**

**1-scrie in citire ceaa ce citeste de la intrarea standard**

**2-inlocuieste toate vocalele cu cifrele 1,2,3,4,5**

**3-afiseaza doar liniile alfanumerice**

#include<sys/types.h>

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<string.h>

#include<wait.h>

#include<dirent.h>

#include<sys/wait.h>

#include<sys/types.h>

#include<sys/stat.h>

int main()

{

int pipe1[2],pipe2[2];

int i,pid,pid2,j,k,q=1;

char sir[256],sir2[256],s[30];

if(pipe(pipe1)<0)

{

printf("eroare la creare pipe1\n");

exit(1);

}

if(pipe(pipe2)<0)

{

printf("eroare la creare pipe2\n");

exit(1);

}

pid=fork();

if(pid<0)

{

printf("eroare la fork\n");

exit(1);

}

if(pid==0)

{

//fiul(1)

close(pipe1[0]);

printf("dati sirul\n");

strcpy(sir,"");

while (strcmp(s,"gata\n")!=0)

{

fgets(s,30,stdin);

if(strcmp(s,"gata\n")!=0)

strcat(sir,s);

}

write(pipe1[1],sir,(strlen(sir)+1));

close(pipe1[1]);

exit(0);

}

else

{

wait(NULL);

close(pipe1[1]);

read(pipe1[0],sir2,100);

close(pipe1[0]);

pid2=fork();

if(pid2<0)

{

printf("Eroare fork2\n");

exit(1);

}

if(pid2==0)

{

//fiul (2)

char v[5]={'a','e','i','o','u'};

for(i=0;i<strlen(sir2);i++)

{

for(j=0;j<5;j++)

if(sir2[i]==v[j])

sir2[i]=j+1+'0';

}

close(pipe2[0]);

write(pipe2[1],sir2,(strlen(sir2)+1));

close(pipe2[1]);

exit(0);

}

wait(NULL);

char alfa[100]="`~!@#$%^&\*()-\_+=[]{};:''"",.<>/?";

close(pipe2[1]);

read(pipe2[0],sir2,256);

close(pipe2[0]);

printf("\n");

char \*linie;

linie=strtok(sir2,"\n");

while(linie!=NULL)

{

q=1;

for(i=0;i<strlen(linie);i++)

for(j=0;j<strlen(alfa);j++)

if(linie[i]==alfa[j])

q=0;

if(q==1)

printf("%s\n",linie);

//else

//printf("nu e alfanumeric\n");

linie=strtok(NULL,"\n");

}

}

}