Process Duplication

Q1. Write a program that creates child process of a process i.e. process P having child process P1

#include<sys/types.h>

#include<unistd.h>

#include<stdio.h>

int main()

{

                pid\_t  p;

                p=fork(); // creates duplicate process P1

                switch(p)

                {

                                case -1:

                                                printf("Error\n");

                                                break;

                                case 0:  // executed by child

                                                printf(“I am child having id:%d\n”,getpid()); // prints pid of process

                                                printf(“My parent’s id is:%d\n”,getppid());  // prints pid of parent process

                                                break;

                                default:  // executed by parent

                                                printf(“I am parent having id:%d\n”,getpid());

                                                printf(“My child’s id is: %d\n”,p);

                                                break;

                }

}

Q2. Write a program to create a child process. The parent process prints 20-29 and child process prints 0-9. Also both the process prints a common message.

#include<sys/types.h>

#include<unistd.h>

#include<stdio.h>

int main()

{

                pid\_t p;

                int i,j;

                p=fork();

                switch(p)

                {

                                case -1:

                                                printf("Error\n");

                                                break;

                                case 0:

                                                for(i=0;i<10;i++)

                                                {

                                                printf("%d\n",i);

                                                sleep(1);

                                                }

                                                break;

                                default:

                                                wait(); //makes the parent wait for the child to finish

                                                for(j=20;j<30;j++)

                                                {

                                                printf("%d\n",j);

                                                sleep(1);

                                                }

                                                break;

                }

printf("common\n"); // Any statement outside switch is executed by both parent and child

}