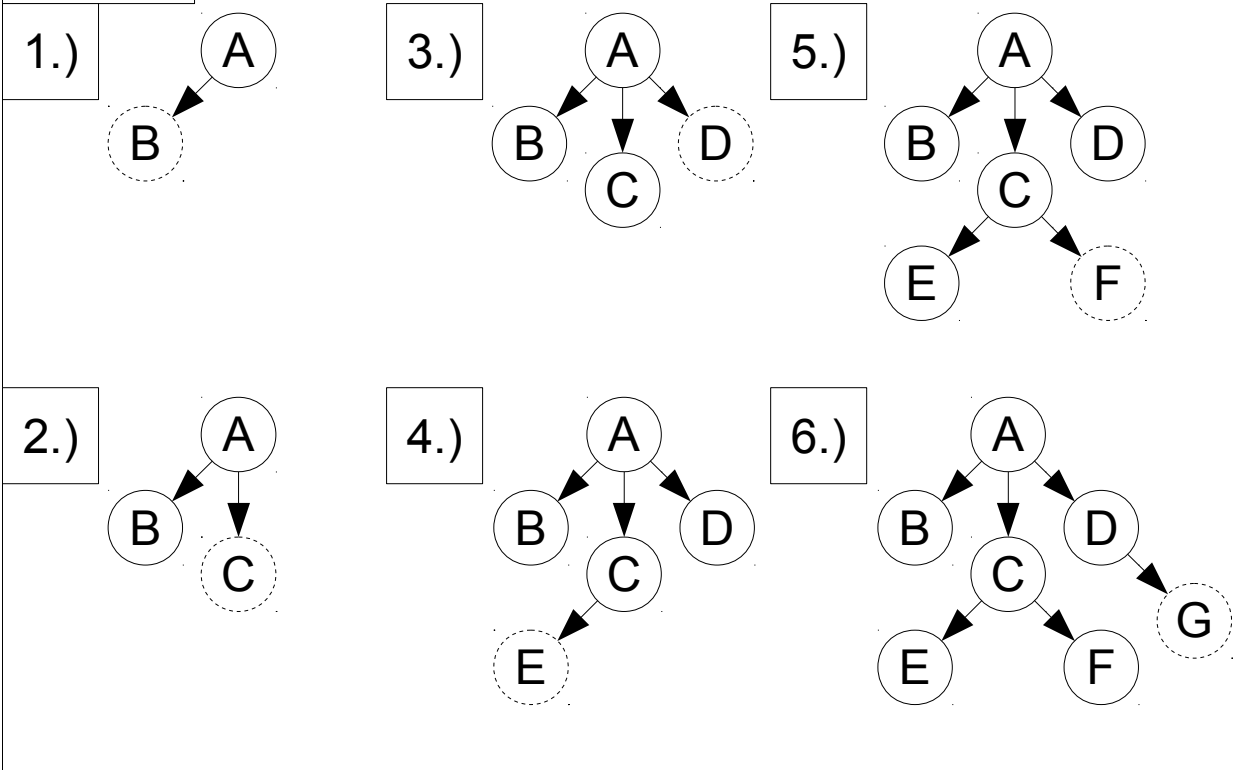
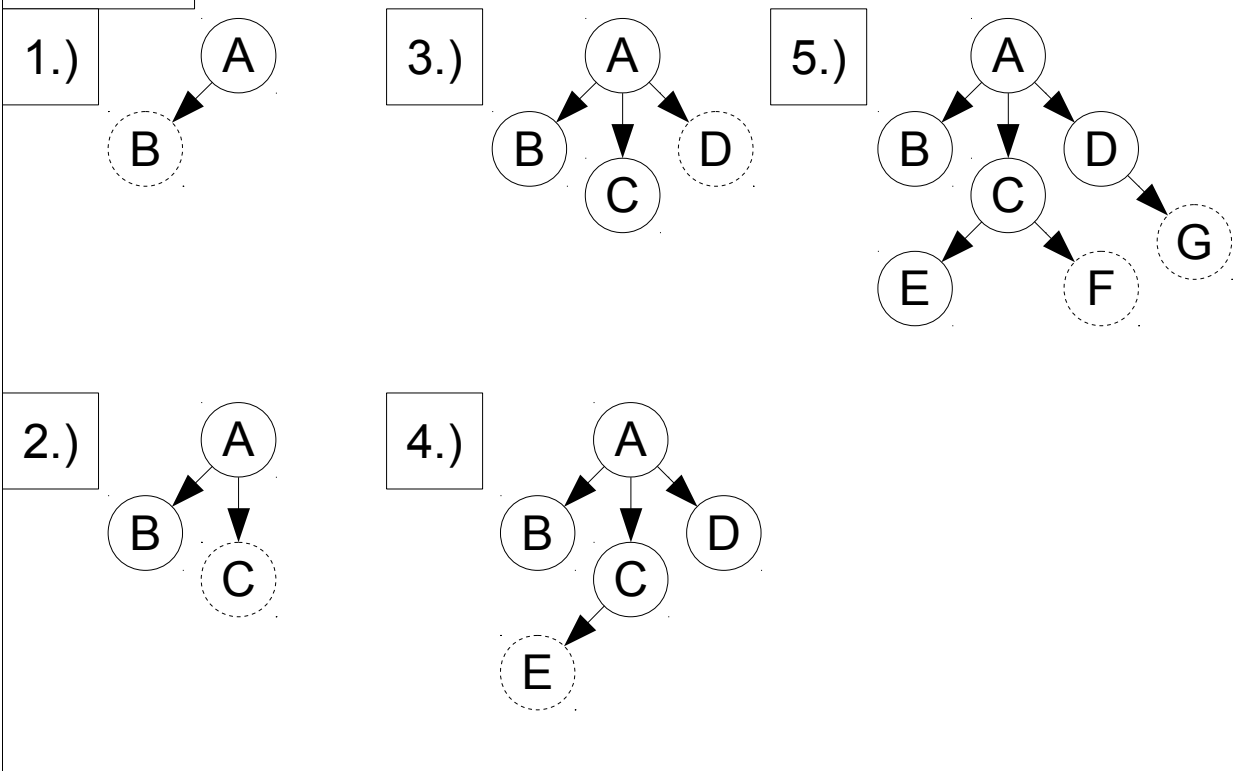


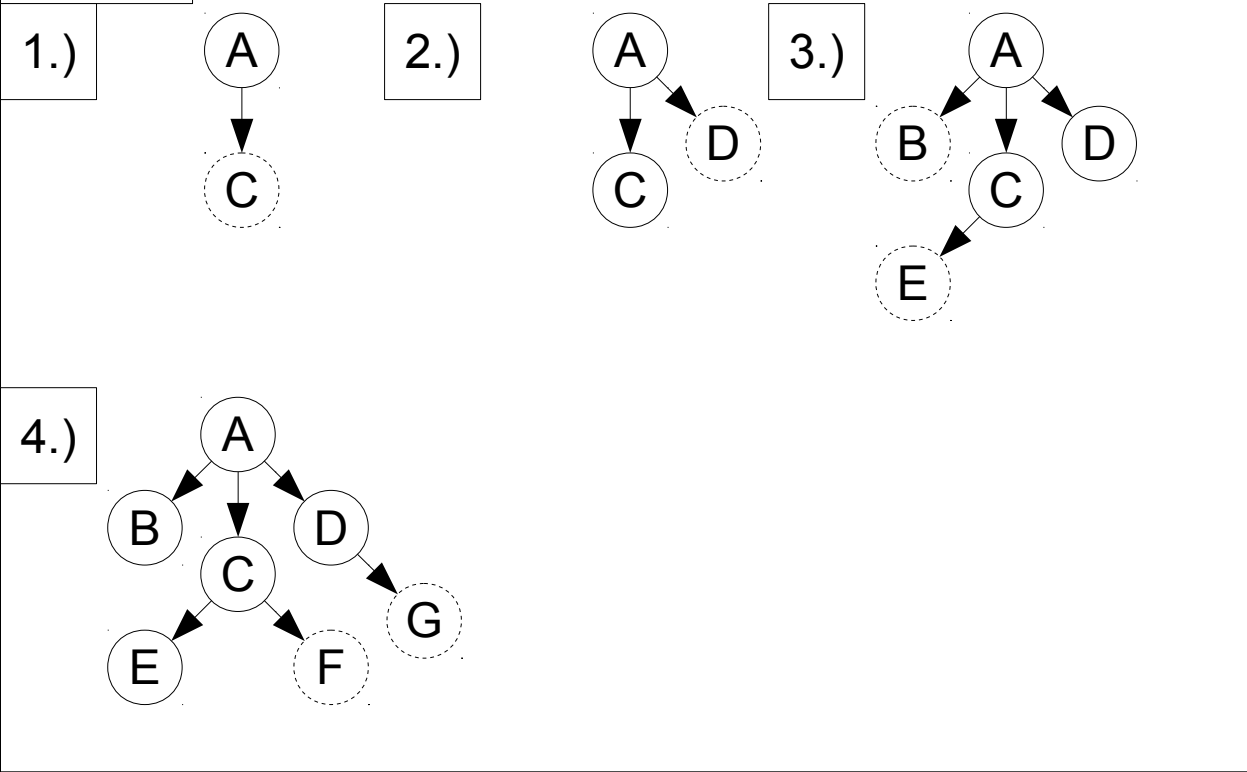
6 Forks:



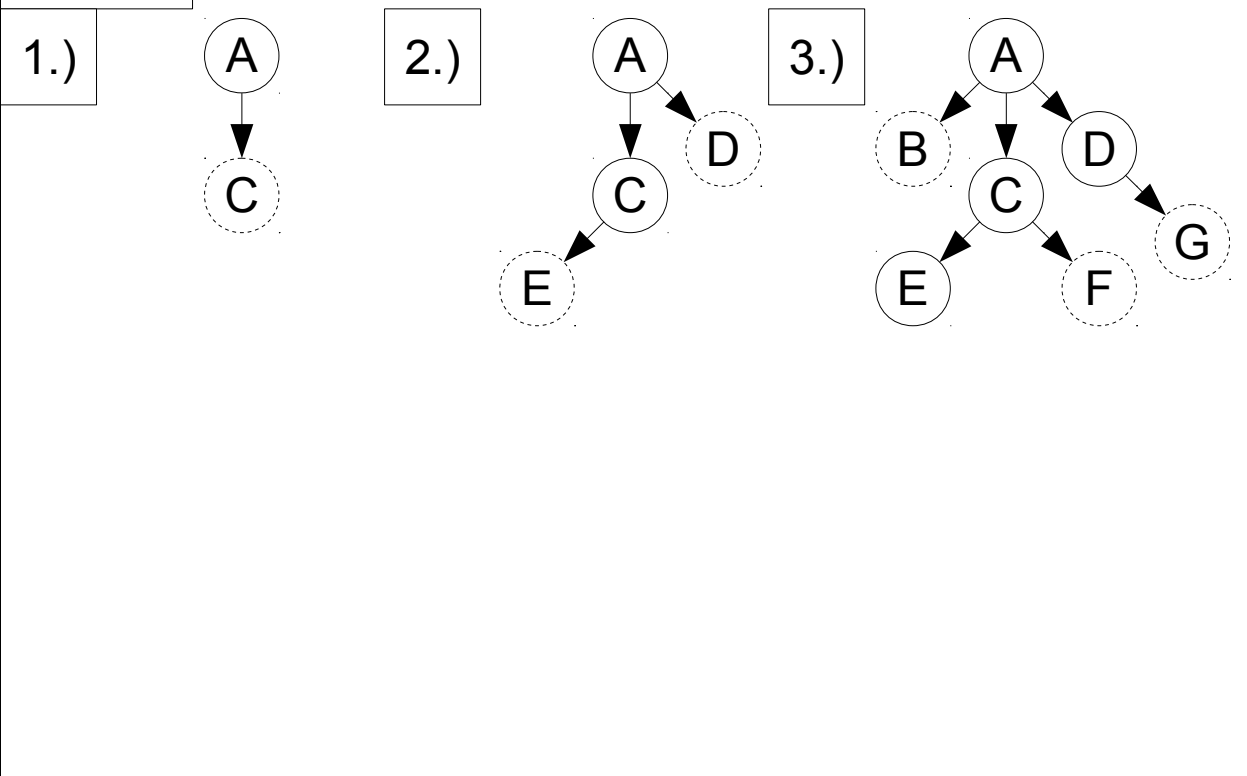
5 Forks:



4 Forks:



3 Forks:



Fork 6:

Process	Fork 1	Fork 2	Fork 3	Fork 4	Fork 5	Fork 6
<b>A</b>	B	C	D	X	X	X
<b>B</b>	X	X	X	X	X	X
<b>C</b>	X	X	X	E	F	X
<b>D</b>	X	X	X	X	X	G
<b>E</b>	X	X	X	X	X	X
<b>F</b>	X	X	X	X	X	X
<b>G</b>	X	X	X	X	X	X

Fork 5:

Process	Fork 1	Fork 2	Fork 3	Fork 4	Fork 5
<b>A</b>	B	C	D	X	X
<b>B</b>	X	X	X	X	X
<b>C</b>	X	X	X	E	F
<b>D</b>	X	X	X	X	G
<b>E</b>	X	X	X	X	X
<b>F</b>	X	X	X	X	X
<b>G</b>	X	X	X	X	X

Fork 4:

Process	Fork 1	Fork 2	Fork 3	Fork 4
<b>A</b>	C	D	B	X
<b>B</b>	X	X	X	X
<b>C</b>	X	X	E	F
<b>D</b>	X	X	X	G
<b>E</b>	X	X	X	X
<b>F</b>	X	X	X	X
<b>G</b>	X	X	X	X

Fork 3:

Process	Fork 1	Fork 2	Fork 3
<b>A</b>	C	D	B
<b>B</b>	X	X	X
<b>C</b>	X	E	F
<b>D</b>	X	X	G
<b>E</b>	X	X	X
<b>F</b>	X	X	X
<b>G</b>	X	X	X

```

/* Fork_Code.c
 * created by Daniel Noyes
 * Will Demonstrate the four various of fork in the bonus lab00
 */

//story should talk about a time machine, reverting the family tree a few times.

#include <unistd.h>
#include <sys/wait.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>

pid_t child[6]; // the Family of A
/*
    child[i]
0:B
1:C
2:D
3:E
4:F
5:G
*/

int CPrint(char a, char *childtype);
pid_t ForkC();
int pidList();

int SixFork();
int FiveFork();
int FourFork();
int ThreeFork();

int main( int argc, char *argv[])
{
    if( argc!= 1)
    {
        printf(" Usage: Fork_Code \n");
        exit(1);
    }

    SixFork();
    FiveFork();
    FourFork();
    ThreeFork();

    return 0;
}
/*
    you did not like how your family was formed, you found a magical bathtub and proceeded to travel
    back in time to rewrite your families history. On the bathtub it said "Please no time paradox, I
    dont want to repair the universe from you stupid mistakes - Doctor" you have no Idea who this doctor
    was but you proceeded without caution
*/

/*
    You proceeded to travel back in time and arange you family nicely
*/

int SixFork()
{
    printf("Time to make a family with 6 forks\n");
    char childtype = 'A';
    //B
    child[0] = ForkC();
    if(child[0] == 0) //New born child
    {
        CPrint('B',&childtype);
    }
}

```

```

        exit(0);
    }
    wait(NULL); //W:B

    //C
    child[1] = ForkC();
    if(child[1] == 0) //New born child
    {
        CPrint('C', &childtype);
        //E
        child[3] = ForkC();
        if(child[3] == 0) //New born child
        {
            CPrint('E', &childtype);
            exit(0);
        }
        wait(NULL); //W:E

        //F
        child[4] = ForkC();
        if(child[4] == 0) //New born child
        {
            CPrint('F', &childtype);
            exit(0);
        }
        wait(NULL); //W:F
        exit(0);
    }
    wait(NULL); //W:C

    //D
    child[2] = ForkC();
    if(child[2] == 0) //New born child
    {
        CPrint('D', &childtype);
        //G
        child[5] = ForkC();
        if(child[5] == 0) //New born child
        {
            CPrint('G', &childtype);
            exit(0);
        }
        wait(NULL); //W:G
        exit(0);
    }
    wait(NULL); //W:D

    return 0;
}

/*
    You Looked at you family and sign, "I just dont know what went wrong" you relized that the family
    was not what you have wished for. Now you looked at the bathtub again and said "well this wont hurt"
    */

int FiveFork()
{
    printf("Time to make a family with 5 forks\n");
    char childtype = 'A';
    //B
    child[0] = ForkC();
    if(child[0] == 0) //New born child
    {
        CPrint('B', &childtype);
        exit(0);
    }
    wait(NULL); //W:B

    //C

```

```

child[1] = ForkC();
if(child[1] == 0) //New born child
{
    CPrint('C',&childtype);
    //E
    child[3] = ForkC();
    if(child[3] == 0) //New born child
    {
        CPrint('E',&childtype);
        exit(0);
    }
    wait(NULL); //W:E
    //F
    goto FiveFork_1; // magically teleport F to pair with D for forking
}
wait(NULL); //W:C

//D
child[2] = ForkC();
if(child[2] == 0) //New born child
{
    CPrint('D',&childtype);
FiveFork_1: ; // for F to fork with D, goto setup
    pid_t pid = ForkC();
    if(pid == 0)
    {
        if(childtype == 'D')
        {
            child[5] = pid;
            CPrint('G',&childtype);
        }
        else
        {
            child[4] = pid;
            CPrint('F',&childtype);
        }
        exit(0);
    }
    wait(NULL); //W:G,F
    if(childtype == 'D')
    {
        child[5] = pid;
    }
    if(childtype == 'C')
    {
        child[4] = pid;
    }

    exit(0);
}
wait(NULL); //W:D
return 0;
}

/*
You look at your family and relized how screwed up everything was. Your family somehow was
unhappy and proceeded to come after you for a aparent reason which you cant figure out. you murmur
"I have to run to the room, where it all started" You just barely go out alive. you relaxed and
tried it again in a smaller fasion.
*/

int FourFork()
{
    printf("Time to make a family with 4 forks\n");
    char childtype = 'A';
    pid_t pid;

    //C

```

```

child[1] = ForkC();//Fork 1
if(child[1] == 0)
{
    CPrint('C',&childtype);
    //E
    goto FourFork_1;
}

//D
child[2] = ForkC();//Fork 2
if(child[2] == 0)
{
    CPrint('D',&childtype);
    goto FourFork_2;
}

wait(NULL);//W:D
//B:0 and E:3

FourFork_1: ;
pid = ForkC();//Fork 3
if(pid == 0)
{
    if(childtype == 'A')
    {
        CPrint('B',&childtype);
    }
    if(childtype == 'C')
    {
        CPrint('E',&childtype);
    }
    exit(0);
}
wait(NULL);//W:B,E
if(childtype == 'A')
{
    child[0] = pid;
}

if(childtype == 'C')
{
FourFork_2: ;
pid = ForkC();//Fork 4
if(pid == 0)
{
    if(childtype == 'D')
    {
        CPrint('G',&childtype);
    }
    if(childtype == 'C')
    {
        CPrint('F',&childtype);
    }
    exit(0);
}
else
{
    if(childtype == 'D')
    {
        child[5] = pid;
    }
    if(childtype == 'C')
    {
        child[4] = pid;
    }
}
wait(NULL);//W:G,F
exit(0);

```

```

    }

    wait(NULL);//C:
    return 0;
}

/*
    You relax as your family was what you wanted. Everything you hoped and dreamed for. One day you
    woke up in a fire, without knowing that one of your children accident set a fire which destroyed
    everything. Such a sad turn of events, you entire family disapeared in a heartbeat. Crawling in the
    cold snow you hear a voice, in Daise you could only see a black figure. A hand reached out and the
    black figure went up to you and whispered "I seen what you did and all what happened, I know you
    pain for It happened to me. I am the Doctor and I warned you for what can happened with sexy(time
    machine). I think I can helpout out this time together where your not alone.
    */

int ThreeFork()
{
    printf("Time to make a family with 3 forks\n");
    char childtype = 'A';
    pid_t pid;

    pid = ForkC();//Fork 1
    if(pid == 0)
    {
        CPrint('C',&childtype);
    }

    pid = ForkC();//Fork 2
    if(pid == 0)
    {
        if(childtype == 'A')
        {
            CPrint('D',&childtype);
        }
        else
        {
            CPrint('E',&childtype);
            exit(1);
        }
    }
    if(childtype == 'C')
    {
        wait(NULL);//wait for E
    }

    pid = ForkC();//Fork 3
    if(pid == 0)
    {
        if(childtype == 'A')
        {
            CPrint('B',&childtype);
        }
        if(childtype == 'C')
        {
            CPrint('F',&childtype);
        }
        if(childtype == 'D')
        {
            CPrint('G',&childtype);
        }
        exit(1);
    }
    wait(NULL);//wait for B,F,G

    if(childtype != 'A') // kills all but A so only C and D
    {
        exit(1);
    }
}

```



```

    }
    wait(NULL); //wait for C
    wait(NULL); //wait for D

    return 0;
}

/*

```

After the grooming events which happened you sit and sign, Right next to you is the same Doctor which helped you with your family. He smiled and said "Well Its my time to go, dont worry I will be back some day. Also here, \*tossed you a simple phone\* if you ever need me again, im on the 1st speed dial.

When walked away you see the bathtub disapear with him. You ponder on what happened and reliezed the events which conspired. You grabed the corn pipe right next to you and sat as the day goes by thanking the mysterious Doctor which helped you on your journey.

```

*/

```

```

/* CPrint(char a)
   Lets go and name our new child in the Family
*/

```

```

int CPrint(char a, char *childtype)
{
    printf("[INFO]: Child(%c) Spawned New Child(%c: %d)\n", *childtype, a, getpid());
    *childtype = a;
    return 0;
}

```

```

/* pid_t ForkC()
   Lets fork and create a new child
   Contains the only fork
*/

```

```

pid_t ForkC()
{
    pid_t pid;

    if((pid = fork()) < 0)
    {
        perror("[Error]: Fork failure");
        exit(1);
    }
    return(pid);
}

```