

Brian Duenas

CSE 460

HW 1

40 points total

1.

Q: How many processes does the following piece of code create? Why?

```
int main()
{
    fork();
    fork();
    fork();
    return 0;
}
```

A: The code creates 8 processes. Each fork creates a parent and child process so, $2*2*2=8$.

2a.

```
//creates a chain of processes
#include <iostream>
#include <sys/types.h>
#include <unistd.h>
#include <sys/wait.h>
#include <stack>

using namespace std;

int main()
{
    pid_t childpid;
    stack<pid_t> oldpid; //old ids

    for (int i = 0; i < 10; ++i) {
        if (childpid = fork())
            break; //parent
        oldpid.push(getppid()); //parent pid
    }
    wait(NULL); //wait
    cout << "This is process " << getpid() << " with child " << childpid <<
        " and ancestors: ";
    while (!oldpid.empty()) { //get (old id) ancestors
        cout << oldpid.top() << " "; //print ancestors
        oldpid.pop(); //remove ancestor
    }
    cout << endl;
    return 0;
} //END
```

Output:

```
This is process 5958 with child 0 and ancestors: 5957 5956 5955 5954 5953 5952 5951 5950
5949 5948
This is process 5957 with child 5958 and ancestors : 5956 5955 5954 5953 5952 5951 5950
5949 5948
This is process 5956 with child 5957 and ancestors : 5955 5954 5953 5952 5951 5950 5949
5948
This is process 5955 with child 5956 and ancestors : 5954 5953 5952 5951 5950 5949 5948
This is process 5954 with child 5955 and ancestors : 5953 5952 5951 5950 5949 5948
This is process 5953 with child 5954 and ancestors : 5952 5951 5950 5949 5948
This is process 5952 with child 5953 and ancestors : 5951 5950 5949 5948
This is process 5951 with child 5952 and ancestors : 5950 5949 5948
This is process 5950 with child 5951 and ancestors : 5949 5948
This is process 5949 with child 5950 and ancestors : 5948
This is process 5948 with child 5949 and ancestors :
```

2b.

```
//fan of processes
#include <iostream>
#include <sys/types.h>
#include <unistd.h>

using namespace std;

int main()
{
    pid_t child;

    for (int i = 0; i < 10; ++i) {
        if ((child = fork()) <= 0)
            break; //child
        sleep(1); //pause/let process run
    }
    cout << "This is child " << getpid() << " with parent " << getppid() << endl;

    return 0;
}
```

Output:

```
This is child 6044 with parent 6043
This is child 6045 with parent 6043
This is child 6046 with parent 6043
This is child 6047 with parent 6043
This is child 6048 with parent 6043
This is child 6049 with parent 6043
This is child 6050 with parent 6043
This is child 6051 with parent 6043
This is child 6056 with parent 6043
This is child 6058 with parent 6043
This is child 6043 with parent 5255
```

3a.

```
//infinite loop
#include <iostream>
#include <sys/types.h>
#include <unistd.h>

using namespace std;

int main()
{
    while (true)
    {
        sleep(1);
    }
}
```

3b.

```
#terminate instances of test1
if ps - ef | grep test1 | grep - v grep - q
then
ps - ef | grep test1 | grep - v grep | awk '{print $2}' | xargs kill &. / dev / null
echo "Process killed"
else
echo "Process Not Running"
fi
```

Output:

```
[brian@csusb.edu@jb359-2 hw1]$ terminateprocess
Process Not Running
[brian@csusb.edu@jb359-2 hw1]$ test1 &
[1] 7552
[brian@csusb.edu@jb359-2 hw1]$ terminateprocess
Process killed
[brian@csusb.edu@jb359-2 hw1]$
```