

```
#include <iostream>
#include <string>
#include <unistd.h>
```

```
using namespace std;
```

```
int main() {
    int fdFirst[2]; // is like a pipe
    // fd[0] is read end and fd[1] where
    // should write
    int fdSecond[2];
    int fdSum[2];
```

```
    string parentMessage = "Parent
Message";
    string childMessage = "Child
Message";
```

```
int sumParent = 0;  
int sumChild = 0;  
int totalSum;
```

```
int i;
```

```
if (pipe(fdFirst) == -1) {  
    cout << "Pipe Parent fail" <<  
endl;  
}; // pipe() creates a pipe
```

```
if (pipe(fdSecond) == -1) {  
    cout << "Pipe Child fail" <<  
endl;  
};
```

```
if (pipe(fdSum) == -1) {
```

```
        cout << "Pipe Sum fail " <<  
endl;  
};
```

```
    int id = fork();  
  
    // once we fork it here file  
discreptor will be copied over,  
therefore it will be 2 fd and if we  
close one fd the other one will still  
be open  
  
    // so fd will be independ from  
each other.  
  
    if (id > 0) {  
        write(fdFirst[1],  
&parentMessage, parentMessage.size());  
        close(fdFirst[0]);  
        close(fdSecond[1]);  
        read(fdSecond[0],  
&childMessage, childMessage.size());
```

```
        cout << "Parent printing  
child's message: " << childMessage <<  
endl;
```

```
        // summation part  
        for(i=1;i<=10;i++)  
            sumParent+=i;  
        read(fdSum[0], &sumChild,  
sizeof(int));
```

```
        totalSum = sumParent +  
sumChild;
```

```
        cout << "Total Sum: " <<  
totalSum << endl;
```

```
    }  
    else if (id == 0) {  
        // here is child  
        read(fdFirst[0],  
&parentMessage, parentMessage.size());
```

```
        cout << "Child printing  
parent's message: " << parentMessage  
<< endl;  
        close(fdFirst[1]);  
        close(fdSecond[0]);  
        write(fdSecond[1],  
&childMessage, childMessage.size());  
        // sumation part  
        for(i=10;i<=20;i++)  
            sumChild+=i;  
        // cout <<sumChild<<endl;  
        write(fdSum[1], &sumChild,  
sizeof(int));  
    }
```

```
    else  
        cout << "Fork is failed" <<  
endl;
```

```
    return 0;  
}
```

```
untitled6 x  
/Users/it/Desktop/os/labs/untitled6/cmake-build-debug/untitled6  
Child printing parent's message: 'Parent Message'  
Parent printing child's message: 'Child Message'  
Total Sum: 220  
  
Process finished with exit code 0
```

