

# IPC (1A)

---

- Inter Process Communication

Copyright (c) 2012 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Please send corrections (or suggestions) to [youngwlim@hotmail.com](mailto:youngwlim@hotmail.com).

This document was produced by using OpenOffice and Octave.

# Pipe (1)

## add100.c

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    float f=0, f100=0;
    char s[10];

    // read string from stdin
    scanf(" %s", s);

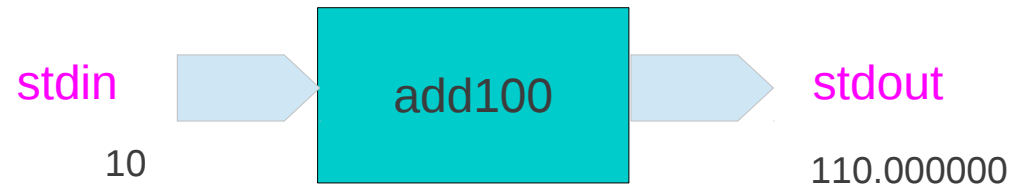
    // convert ascii string to float
    f = atof(s);
    f100 = f + 100;

    // write result to stdout
    printf("%f\n", f100);

    return 0;
}
```

```
gcc -o add100 add100.c
```

```
$. /add100
10          ← input
110.000000 → output
```



# Pipe (2)

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[])
{
    float f=0, f3=0;
    char s[10];

    // read string from stdin
    scanf(" %s", s);

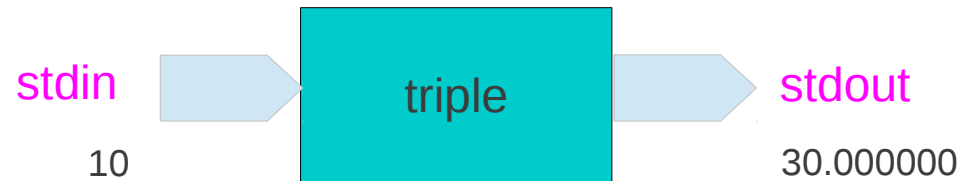
    // convert ascii string to float
    f = atof(s);
    f3 = 3*f;

    // write result to stdout
    printf("%f\n", f3);

    return 0;
}
```

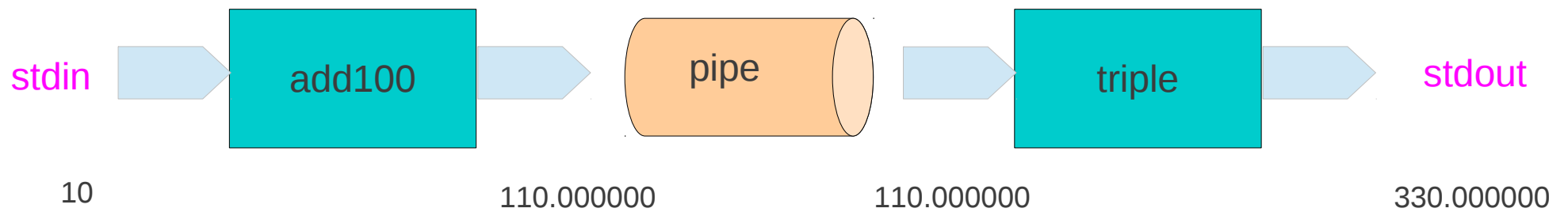
```
gcc -o triple triple.c
```

```
$. /triple
10          ← input
30.000000  → output
```



# pipe (3)

`./add100 | ./triple`



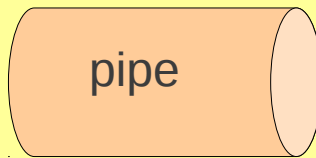
# popen (1)

```
FILE *fp;  
fp = popen("./triple", "w");
```

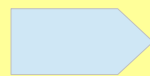
parent

```
fprintf(fp, "10");  
fclose(fp);
```

10



pipe



child

```
triple
```

stdout

30.000000

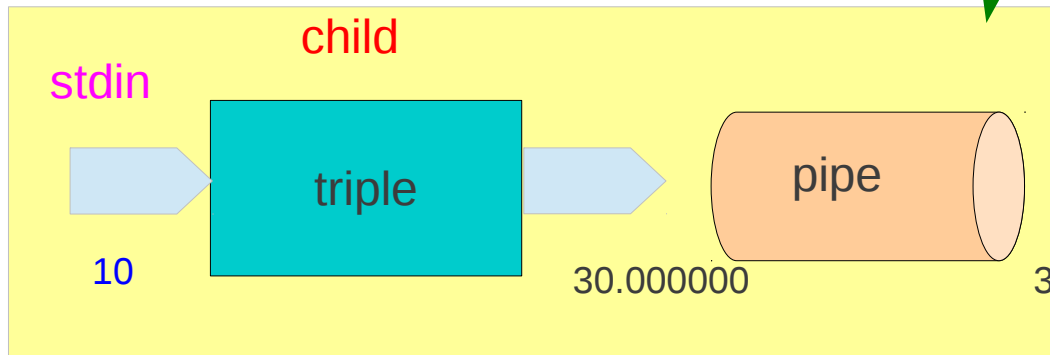
```
FILE *fp;  
fp = popen("./triple", "r");
```

parent

```
fscanf(fp, "%s", str);  
printf("%s\n", str);  
fclose(fp);
```

stdout

30.000000



child

stdin

triple

10

30.000000

pipe

30.000000

## popen (2)

```
int main(int argc, char* argv[])
{
    char inbuf[100];

    FILE *fp;

    // fp = popen("./triple", "w");
    // fprintf(fp, "10");
    // fclose(fp);

    fp = popen("./triple", "r");
    fscanf(fp, "%s", inbuf);
    printf("%s\n", inbuf);
    fclose(fp);

    return 0;
}
```

# popen (3)

---

who | sort

popen, fclose

HW #2



# Reference

---

## References

- [1] <http://en.wikipedia.org/>
- [2] S.S. Park, Linux Practical Command Bible (in Korean)
- [3] S.S. Park, Linux Server Practical Administration Bible (in Korean)