Name:

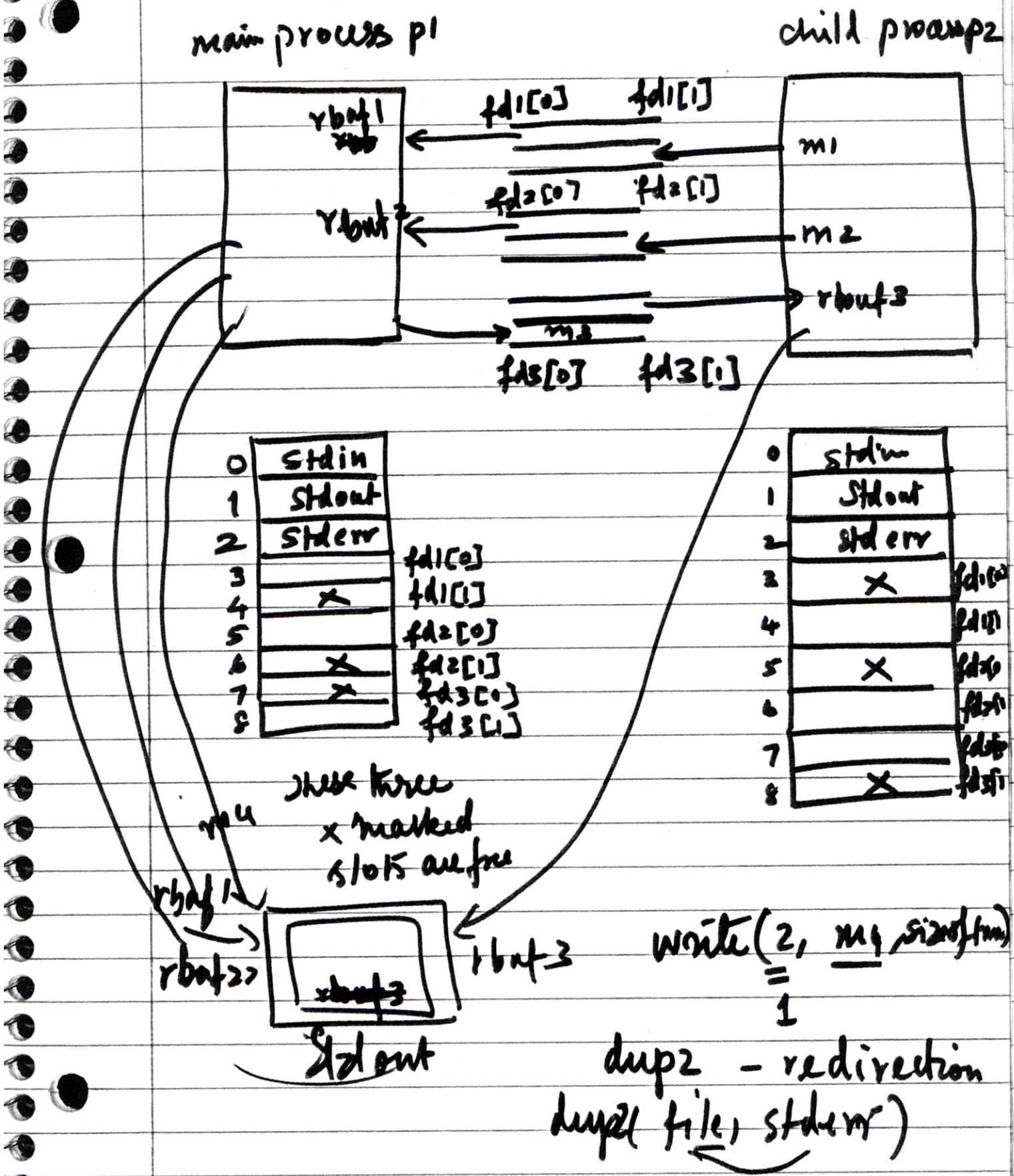
3. (15 points) Task creation and communication using pipes

a) Draw the tasks and the pipes between them that are described by the following code segment.

b) Draw the task descriptor table with the relevant points and index numbers.

c) What is the output?

```
pid_t newpid;
int fd1[2], fd2[2], fd3[2];

char m0[] = "\n about to fork ..\n";
char m1[] = "message 1\n";
char m2[] = "message 2\n";
char m3[] = "message 3\n";
char m4[] = "\n done ..\n";

char rbuf1[256];
char rbuf2[256];
char rbuf3[256];
int cn1, cn2, cn3;

if ((pipe(fd1)== -1)) printf(" error \n");
if ((pipe(fd2)== -1)) printf(" error \n");
if ((pipe(fd3)== -1)) printf(" error \n");

if ((newpid = fork()) == -1) { printf("error \n"); return 0;}

if (newpid > 0) {
  close(fd1[1]);
  close(fd2[1]);
  close(fd3[0]);
  write(8, m3, sizeof(m3));
  usleep(10000);
  cn1 = read(3, rbuf1, 256);
  cn2 = read(5,rbuf2,256);
  write(1, rbuf1, cn1);
  write(1, rbuf2, cn2);

 }
 else {
   close (fd1[0]);
   close (fd2[0]);
   close (fd3[1]);
   write(4, m1, sizeof(m1));
   usleep(10000);
   write(6,m2,sizeof(m2));
   usleep(10000);
   cn3 = read(7, rbuf3, 256);
   write(1, rbuf3,cn3);
 }
 usleep(10000);
 write(2,m4,sizeof(m4));

 return 0;
}
```


main process p1
child process p2
rbuf1
fd1[0] fd1[1]
m1
fd2[0] fd2[1]
rbuf2
m2
rbuf3
m3
fd3[0] fd3[1]
0 stdin
1 stdout
2 stderr
3
4 X
5
6 X
7 X
8
fd1[0]
fd1[1]
fd2[0]
fd2[1]
fd3[0]
fd3[1]
these three x marked slots are free
rbuf1
rbuf2
rbuf3
stdout
write(2, m1, sizeof(m1))
= 1
dup2 - redirection
dup2(file, stderr)