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?

```c
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) {
    // parent will not write into fd1
    close(fd1[1]);
    close(fd2[1]);
    close(fd3[0]);
    write(0, m3, sizeof(m3));
    usleep(100000);
    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(100000);
    write(6, m2, sizeof(m2));
    usleep(100000);
    cn3 = read(7, rbuf3, 256);
    write(1, rbuf3, cn3);
}
usleep(100000);
write(2, m4, sizeof(m4));

return 0;
}
```
<table>
<thead>
<tr>
<th></th>
<th>Stdin</th>
<th>Stdout</th>
<th>Stderr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Check three
marked
$1/15$ are free

rbuf1

rbuf2

main process p1

```

```
<table>
<thead>
<tr>
<th></th>
<th>fd[0]</th>
<th>fd[1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>fd[0]</td>
</tr>
<tr>
<td>4</td>
<td>fd[1]</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>fd[2]</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>fd[3]</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>fd[4]</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>fd[5]</td>
<td></td>
</tr>
</tbody>
</table>

write(2, my_xbuff, 1)

dup2 - redirection
dup2($file$, stderr)
```