# Упражнение 7

1. Да се пуснат един след друг два системни примитива fork(), като за всеки генериран процес се отпечатва pid-а на процеса/getpid()/ и pid-a на родителския процес/getppid()/.
2. Да се пусне един fork() и само в детския процес още един fork(). За всеки генериран процес се отпечатва pid-а на процеса/getpid()/ и pid-a на родителския процес/getppid()/.
3. Да се пуснат три fork() един след друг. Колко процеса се генерират?
4. Да се създадат 2 процеса. В детския се пише Hello на стандартния изход. След това се създават още 2 процеса и се пише Hello1 на стандартния изход. Колко пъти излиза Hello и колко Hello1?
5. Да се напише програма на С, която получава като параметри две имена на файлове. Отваря първия файл за писане (ако не съществува, го създава, в противен случай го занулява), създава 2 процеса. Родителят пише hello1, а детето – hello2 във първия файл.
6. В зад 4 пренасочвате изхода на детето във втория файл. В общи действия на двата процеса се записват hello3 на стандартния изход и hello4 във втория файл:
   * с wait в родителя
   * без wait в родителя
7. В зад 5 в края на локалните действия на детето възстановявате стандартния му изход.
8. В зад 6 – да се пробва с exit() – в единия или двата процеса; с или без wait()
9. Да се напише програма на C, която получава като параметри от командния ред две команди (без параметри). Изпълнява първата. Ако тя е завършила успешно изпълнява втората. Ако не, завършва с код -1.
10. Да се напише програма на C, която получава като командни параметри две команди (без параметри). Изпълнява ги едновременно и извежда на стандартния изход номера на процеса на първата завършила успешно. Ако нито една не завърши успешно извежда -1
11. Трасировка – upr7\_tras1; upr7\_tras2;