

## Process\_07

```
C process_07.c 3 x
C process_07.c > ...
1  #include <stdio.h>
2  #include <unistd.h>
3  #include <sys/types.h>
4  #include <sys/wait.h>
5  #include <stdlib.h>
6  #define SLEEP_TIME 2
7
8  int main(int argc, char* argv[]){
9      pid_t pid;
10     int status;
11     int parent_pid, child_pid, favorite_number;
12     char favorite_fruit[] = "apple";
13
14
15     printf("Final Question!\n");
16
17     pid = fork();
18
19     //you are allowed to write your own printf
20     switch(pid) {
21         default:
22             parent_pid = getpid();
23             printf("[%d] I am a parent\n", parent_pid);
24             wait(&status);
25             printf("[%d] ....and my child's favorite number is %d\n", parent_pid, status >> 8);
26             //hint: how would this process get the result of another??
27             break;
28         case 0:
29             sleep(SLEEP_TIME);
30             favorite_number = 5;
31             child_pid = getpid();
32             printf("[%d] and I am a child!\n", child_pid);
33             printf("[%d] my parent's favorite fruit is %s but he doesn't know that my favorite number is %d\n", child_pid, favorite_fruit, favorite_number);
34             exit(favorite_number);
35             break;
36         case -1:
37             printf("and I am ???");
38             break;
39     }
40
41
42     return 0;
43 }
```

설명:

Line 1~5: 필요한 라이브러리 헤더 포함. Printf, fork, sleep, getpid wait, exit 사용을 위함.

Line 6: 자식 프로세스가 대기할 sleep의 시간을 상수로 정의.

Line 8: main 함수 선언

Line 9~12: 사용 변수 선언

- pid를 저장할 pid
- wait의 결과를 저장할 status
- 부모 프로세스의 pid를 저장하는 parent\_pid
- 자식 프로세스의 pid를 저장하는 child\_pid
- 자식이 좋아하는 숫자 favorite\_number
- 부모가 좋아하는 과일 favorite\_fruit

Line 15: 마지막 문제 시작 문자

Line 17, 20: pid=fork로 fork 함수 호출(부모 자식 프로세스) 및 switch 조건을 pid로 설정

Line 21~27: 부모를 default로 설정하여 부모 분기 내용 코드

부모의 pid를 가져와 parent\_pid에 저장, 부모라는 메시지 출력, wait로 자식 프로세스의 종료를 기다린 뒤 자식이 반환한 status를 8bit shift(원래 value)한 뒤 그것을 자식이 가장 좋아하는 number로 출력.

Line 28~35: 자식(pid = 0 케이스) 분기 코드

SLEEP\_TIME만큼 sleep로 대기, 자식이 가장 좋아하는 숫자를 5로 설정.

자식의 pid를 child\_pid에 저장. 내가 자식이라는 메시지 출력.

부모가 가장 좋아하는 과일과 자신이 가장 좋아하는 숫자에 대한 메시지 출력.

Exit를 통해 가장 좋아하는 숫자를 종료 상태로 부모에게 반환

Line 36~38: fork 실패 시 분기 내용(에러 코드)

Line 42: .메인 함수 종료 return 0

이번 과제에서는 getpid와 wait, sleep, exit 총 4개의 실습 코드를 이용했습니다.

```
kuc@2021320057:/mnt/shared_folder/process$ ./07
Final Question!
[7934] I am a parent
[7935] and I am a child!
[7935] my parent's favorite fruit is apple but he doesn't know that my favorite number is 5
[7934] ....and my child's favorite number is 5
kuc@2021320057:/mnt/shared_folder/process$
```

실행 완료 코드