```
Code 1:
Int main() {
pid_t pid = fork();
if (pid == -1) {
std::cerr << "Error in fork." << std::endl;
return 1;
}
if (pid == 0) {
// Child process
std::cout << "Child process." << std::endl;
exit(0);
} else {
// Parent process
std::cout << "Parent process." << std::endl;
wait(NULL);
}
return 0;
}
```

```
Code 2:
int main() {
for (int i = 0; i < 3; ++i) {
pid_t pid = fork();
if (pid == -1) {
std::cerr << "Error in fork." << std::endl;
return 1;
}
if (pid == 0) {
// Child process
std::cout << "I am child" << i << std::endl;
exit(0);
} else {
// Parent process
wait(NULL);
}
}
return 0; }
```

```
Code 3
```

```
int main() {
if (fork() ll fork()) {
std::cout << "Process Created" <<std::endl;
} else {
wait(NULL);
}
return 0;
}
```

```
Code 4:
int main() {
if (fork() && fork()) {
std::cout << "Process Created" <<std::endl;
} else {
wait(NULL);
}
return 0;
}
```

```
Code 5:
int main() {
if (fork() ll (fork() && fork()) ll fork()) {
std::cout << "Process Created" <<std::endl;
} else {
wait(NULL);
}
return 0;
}
```

```
Code 6:
int main() {
if (fork() && (fork() ll fork())) {
std::cout << "Process Created" <<std::endl;
} else {
wait(NULL);
}
return 0;
}
```