Assignment 1 Marking Schema

For Q1.1:

The test case is ./q11.out 3 2 30

```
This is the BEGINNING of the program.

Apply 12 bytes.
Parent process ID: 44481. order 1
Differences: -1, 28.  +2 marks (correctly use shared mem)
Send a SIGCONT to process 44482.

Received a SIGCONT.
Child process ID: 44482. order 2
Sum of differences: 27. +2 marks (correctly use shared mem)

Exited Process ID: 44482. +3 marks if == child ID    order 3
This is the END of the program. +3 marks Only 1 END is printed (correctly exit)
```

Three order is correct. +10 marks
Noted that if you do not print the id or you do not correctly spell "parent process ID" (case insensitive), you will get 0 mark for this part.

Noted that we have check whether the number of child created by fork is correct. If it is not correct, your grade of Q1.1 = Q1.1 * 50%

For Q1.2:

The test case is ./q12.out 3 2 30

```
This is the BEGINNING of the program.

Apply 12 bytes.
Parent process ID: 44591. order 1
Differences: -1, 28.  +0.5 marks
Send a SIGCONT to process 44593.

Received a SIGCONT.
Child process A ID: 44593. order 2
Sum of differences: 27. +0.5 marks
Send a SIGCONT to process 44592.

Received a SIGCONT.
Child process B ID: 44592. order 3
The 3rd argument is larger than the 1st argument. +1 marks

                count is correctly printed +1.5
Exited Process ID: 44593; Count: 1. order 4
Exited Process ID: 44592; Count: 2.  order 5
This is the END of the program. +1.5 marks Only 1 END is printed (correctly exit)
```

Five order is correct. +10 marks
Noted that if you do not print the id or you do not correctly spell "parent process ID" (case insensitive), you will get 0 mark for this part.

Noted that we have check whether the number of child created by fork is correct. If it is not correct, your grade of Q1.2 = Q1.2 * 50%

Q2.1

The test case is ./q21.out 1 9

- 6 marks: 4 processes created in total
- 2 marks: sorting result is correct
- 2 marks: shmdt() is correctly used

Q2.2

The test case is ./q22.out 3 4

- 21 marks: 64 processes created in total (most important part, you do recursively fork correctly)
- 10.5 marks: the sorting result is correct (correctly use waitpid)
- 3.5 marks: shmdt() is correctly used