

과제 #01

확장형 고성능 컴퓨팅 (2021년도 2학기, M1522.006700, M3239.002300)

Due: 2021년 9월 26일(일) 23시 59분

1 Compilation Process

`sqrt.c`는 실수 하나를 입력받아 제곱근을 출력하는 간단한 C 코드이다. 다음 질문들에 답하라.

1.1 Preprocessing

- (a) (5점) `sqrt.c`는 두 개의 헤더 파일(`stdio.h`와 `math.h`)을 `include`한다. 하지만 우리는 이 헤더 파일들을 작성한 적이 없다. 이 두 파일은 어디에 있는가? 파일을 찾아서 열어보고, 정확한 파일 경로와 파일의 줄 수를 답하라.
(Hint: `cpp -v /dev/null`로 C preprocessor인 `cpp`의 설정들을 확인할 수 있다.)
- (b) (5점) Preprocess까지만 진행하는 `gcc` 옵션을 찾아서 `sqrt.c`를 preprocess하라. Preprocess 된 결과에서 `scanf`, `printf`, `sqrt`를 각각 찾아서 해당 부분을 보고서에 붙여넣어라.
(Hint1: `man gcc`로 `gcc`의 옵션들을 확인할 수 있다. `preprocessing`으로 검색하면 쉽게 찾을 수 있다.)
(Hint2: `gcc`는 preprocessing 결과를 `stdout`, 즉 화면에 직접 출력한다. 이를 파일에 저장하고 싶다면 `-o` 옵션이나 리다이렉션(`>`)을 사용하자.)
- (c) (5점) Preprocess 된 결과에 `scanf`, `printf`, `sqrt`의 실제 구현이 들어있는가? 다시 말해, `sqrt` 코드에 실제로 제곱근을 구하는 코드가 포함되어 있는가? 포함되어 있다면 코드를 간략하게 설명하고, 포함되어 있지 않다면 포함되어 있지 않은 이유를 답하라.

1.2 Compilation

- (a) (5점) Object file을 출력하는 `gcc` 옵션을 찾아서 `sqrt.o`를 생성하라. 생성하는데 사용한 명령을 답하라.
- (b) (5점) Preprocessing의 결과는 C 코드가 담긴 단순한 텍스트 파일이었다. `sqrt.o`의 파일 포맷은 무엇인가? 파일 포맷과 파일 포맷을 알아낸 방법을 답하라.

1.3 Linking

- (a) (10점) `sqrt.o`를 아무런 옵션 없이 컴파일을 시도하면 다음과 같은 에러가 발생한다.

```
$ gcc sqrt.o
sqrt.o: In function 'main':
sqrt.c(.text+0x3d): undefined reference to 'sqrt'
collect2: error: ld returned 1 exit status
```

왜 이러한 에러가 발생하는가? 이유와 함께 올바르게 컴파일하여 최종 실행파일 `sqrt`를 생성하는 명령을 답하라.

- (b) (5점) `sqrt`를 실행하여 임의의 숫자를 입력해보고, 결과 화면을 캡처하여 보고서에 첨부하라.

2 C Programming Practice

2.1 Prime Factorization (20점)

2 이상 10000 이하의 자연수를 입력으로 받아 소인수분해하는 프로그램 `prime.c`을 작성하라. 수가 한 개의 소수로 여러 번 나누어 떨어진다면 나누어 떨어지는 횟수만큼 출력해야 하며, 결과는 오름차순으로 정렬되어야 한다. 조교가 작성한 `ref/prime`과 실행 결과가 같으면 된다. 실행 예시는 다음과 같다.

```
$ gcc prime.c -o prime
$ ./prime
2
2
$ ./prime
24
2 2 2 3
$ ./prime
9999
3 3 11 101
```

2.2 Vortex (20점)

1 이상 10 이하의 자연수 N 을 입력으로 받아 $N \times N$ 크기의 소용돌이 문자 배열을 출력하는 프로그램 `vortex.c`를 작성하라. 소용돌이 문자 배열이란, 바깥쪽에서부터 시계방향으로 돌면서 알파벳 대문자를 오름차순으로 채운 배열이다. 알파벳을 Z까지 모두 사용한 경우 다시 A로 돌아가서 반복된다. 조교가 작성한 `ref/vortex`와 실행 결과가 같으면 된다. 실행 예시는 다음과 같다.

```
$ gcc vortex.c -o vortex
$ ./vortex
5
ABCDE
PQRSF
OXYTG
N WVUH
MLKJI
$ ./vortex
6
ABCDEF
TUVWXG
SFGHYH
REJIZI
QDCBAJ
PONMLK
```

2.3 Selection Sort (20점)

Selection sort는 N 개의 정수가 주어졌을 때, 최소값을 찾아 앞으로 옮기는 과정을 $N - 1$ 회 반복하여 정렬하는 알고리즘이다. 뼈대코드 `selection.c`의 비어있는 부분을 완성하라. 조교가 작성한 `ref/selection`과 실행 결과가 같으면 된다. 실행 예시는 다음과 같다.

```
$ gcc selection.c -o selection
$ ./selection
10
23 -3 5 9 11 33 87 -7 -24 50
```

-24 -3 5 9 11 33 87 -7 23 50
-24 -7 5 9 11 33 87 -3 23 50
-24 -7 -3 9 11 33 87 5 23 50
-24 -7 -3 5 11 33 87 9 23 50
-24 -7 -3 5 9 33 87 11 23 50
-24 -7 -3 5 9 11 87 33 23 50
-24 -7 -3 5 9 11 23 33 87 50
-24 -7 -3 5 9 11 23 33 87 50
-24 -7 -3 5 9 11 23 33 50 87

3 Submission Instruction

- `report.pdf`, `prime.c`, `vortex.c`, `selection.c`를 한 파일로 압축하여 ETL에 제출한다.
- 첨부 파일명은 계정이름_HW01.zip으로 한다. (e.g., shpc99_HW01.zip)
- 본인의 계정이름을 모를 경우 ETL 자료실에 업로드 되어있는 게시글에서 확인한다.
- 채점은 프로그램에 의해 기계적으로 처리되므로 위 사항을 지키지 않은 경우 누락되거나 불이익을 받을 수 있다.
- Grace day를 사용하려면 본인이 숙제를 제출한 날에 조교에게 메일(shpc21@aces.snu.ac.kr)로 알려야 한다. 메일 없이 제출만 한 경우 다음 숙제를 위해 아낀 것으로 판단, 미제출 처리된다.