

1.

1.1

(a)

제시된 명령어로

#include <...> search starts here:

/usr/lib/gcc/x86_64-linux-gnu/9/include

/usr/local/include

/usr/include/x86_64-linux-gnu

/usr/include

가 포함됨을 알 수 있습니다. 해당 경로에는

/usr/include/stdio.h

/usr/include/x86_64-linux-gnu/bits/stdio.h

/usr/include/c++/9/tr1/stdio.h

가 사용되고

/usr/include/ucs/sys/math.h

/usr/include/math.h

/usr/include/c++/9/tr1/math.h

/usr/include/c++/9/math.h

를 통하여 usr/include에서 가져옴을 알 수 있습니다.

(b)

-E 옵션을 주면 됩니다.

gcc -E sqrt.c

```
extern double sqrt (double __x) __attribute__ ((__nothrow__ , __leaf__)); extern double __sqrt (double __x) __attribute__ ((__n  
  
extern int printf (const char *__restrict __format, ...);
```

```

extern int scanf (const char *__restrict __format, ...) ;
extern int sscanf (const char *__restrict __s,
    const char *__restrict __format, ...) __attribute__ ((__nothrow__ , __leaf__));

extern int fscanf (FILE *__restrict __stream, const char *__restrict __format, ...) __asm__ ("" "__isoc99_fscanf")
;
extern int scanf (const char *__restrict __format, ...) __asm__ ("" "__isoc99_scanf")
;

```

(c)

포함이 되어있습니다.

```

extern double sqrt (double __x) __attribute__ ((__nothrow__ , __leaf__)); extern
double __sqrt (double __x) __attribute__ ((__nothrow__ , __leaf__));

```

과 같은 코드는

extern 을 통하여 이 함수는 외부에 정의되어있음을 알려줍니다. 다른 함수들도 다른 라이브러리에서 끌어와야함을 알려줍니다.

1.2

(a)

-c 옵션을 사용합니다.

```
gcc -c sqrt.c -o sqrt.o
```

(b)

File 명령어를 활용합니다.

64비트 elf 포맷으로 명령어는 file sqrt.o가 됩니다.

1.3

(a)

-lm 옵션을 주어야합니다.

해당 코드에서 사용하려는 math라이브러리를 포함하기 위함입니다.

비슷하게 -lpthread등이 있습니다.

(b)

```
shpc033@login0:~/hw1$ ./sqrt 16
4.00000000
shpc033@login0:~/hw1$ ./sqrt 13
3.60555128
shpc033@login0:~/hw1$
```

3.

(a)

```
shpc033@login0:~/hw1$ sinfo
```

PARTITION	AVAIL	TIMELIMIT	NODES	STATE	NODELIST
class1	up	1:00	1	mix	a00
class1	up	1:00	11	idle	a[01-11]

첫줄은 현재의 한 노드가 mix 상태로 실행중임을

두번째는 11개의 노드가 쉬는 중임을 나타냅니다.

(b)

JOBID	PARTITION	NAME	USER	ST	TIME	NODES
NODELIST(REASON)						

22296	class1	convert	shpc049	R	0:30	1 a00
-------	--------	---------	---------	---	------	-------

해당 아웃풋은 현재 클래스1 소속 49번 학생의 컨버트 작업이 대기중입니다.

(c)

```
shpc033@login0:~/hw1$ srun -p class1 -N 2 hostname
```

```
a00
```

```
a01
```

명령을 실행한 두개의 노드를 출력합니다.(호스트네임 명령어가 출력)

(d)

Cpu의 구성을 알아낼수 있는 lscpu를 로컬과 srun을 통한 다른머신을 통하여 실행함으로써 두 명령어에 대하여 다른 종류의 씨피유가 출력이 됩니다.

```
umip rdpid overflow_recov succor smca
● shpc033@login0:~/hw1$ srun -p class1 -N 1 lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:             Little Endian
Address sizes:          46 bits physical, 48 bits virtual
CPU(s):                 64
On-line CPU(s) list:    0-63
Thread(s) per core:     2
Core(s) per socket:     16
Socket(s):              2
NUMA node(s):           2
Vendor ID:              GenuineIntel
CPU family:              6
Model:                  85
Model name:             Intel(R) Xeon(R) Silver 4216 CPU @ 2.10GHz
Stepping:               7
CPU MHz:                800.001
CPU max MHz:            3200.0000
CPU min MHz:            800.0000
BogoMIPS:               4200.00
Virtualization:         VT-x

● shpc033@login0:~/hw1$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:             Little Endian
Address sizes:          43 bits physical, 48 bits virtual
CPU(s):                 128
On-line CPU(s) list:    0-127
Thread(s) per core:     2
Core(s) per socket:     32
Socket(s):              2
NUMA node(s):           2
Vendor ID:              AuthenticAMD
CPU family:              23
Model:                  49
Model name:             AMD EPYC 7502 32-Core Processor
Stepping:               0
Frequency boost:        enabled
CPU MHz:                1487.614
CPU max MHz:            2500.0000
CPU min MHz:            1500.0000
BogoMIPS:               4999.74
Virtualization:         AMD-V
L1d cache:              2 MiB
```