Quiz A Module 3

Due Apr 17 at 11:59pm **Points** 8 **Questions** 8
Time Limit 10 Minutes

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	10 minutes	8 out of 8

Submitted Apr 12 at 7:16pm

Question 1

1 / 1 pts

```
char *text;

strcpy(text,hello);

//Thats ok?
```

Correct!

● Very possible a segfault or other problems.

○ Totally fine.

Question 2

1 / 1 pts

```
int a=5;

int* i,p;

p=&a;

*p=15;
```

```
printf("%d",a);
```

What will be printed?

Correct!

● That's a compiler error.

○ 0

○ 5

○ 15

Question 3

1 / 1 pts

The program break at the beginning of a process is

Correct!

● At an unknown address, which can be received with sbrk(0);

○ At 0!

Question 4

1 / 1 pts

We are in a modern OS with virtual memory.
Lets say you are able to write directly into address 1:

```
int *p = 1;
//and write something in it:
*p = 12345;
//what are you overwriting?
```

○ The OS Kernel code!

○ The program stack!

○ The heap.

Correct!

● The program executable code itself

Question 5

1 / 1 pts

int *p = 0x00000E00; //Given, that your compiler is fine with direct addressing

p = p + 1;

//Whats p's address now?

○ 0x00000E00

○ 1x00000E01

Correct!

● 0x00000E04

○ 0x00000E02

○ 0x00000E01

Question 6

1 / 1 pts

unsigned char *p = 0x00000E01; //Given, that your compiler is fine with direct addressing

p = p + 1;

//Whats p's address now?

○ 0x10000E01

Correct!

● 0x00000E02

○ 0x00000E05

○ 1x00000E01

○ 0x00000E00

Question 7

1 / 1 pts

```
clock_t start, stop;

start = clock();

//your algo goes here

stop = clock();

printf("%f",(double)(start - stop));

//This is printing ...
```

○ ... the seconds since the start of the computer.

○ ... the time passed during the algorithm.

Correct!

● ... the inverse of the time passed during the algorithm.

Question 8

1 / 1 pts

```
void *p = mmap(NULL, 4, PROT_READ | PROT_WRITE,
MAP_SHARED, -1, 0);
```

//Turns out "p" holds an invalid address! Why?

○ Both flags read AND write are active which is mutually exclusive!

Correct!

● MAP_ANONYMOUS is missing (because its purpose is to allocate free memory, not store a file)

○ To hell with mmap.

○ The first argument must be an address in the free region of the process address space, and not NULL!