

# CS 519 - C Prep Quiz

---

*Developed by Prof. Taesoo Kim, Georgia Tech*

Name:

NetID:

## 1) What is the output of the following code snippet?

```
uint32_t v = 0xdeadbeef;  
printf("%02x", (char *)v[0]);
```

- 1) de
- 2) ad
- 3) be
- 4) ef
- 5) 00

ANSWER:

## 2) What is the output of the following code snippet?

```
int64_t v = 0xdeadbeef;  
printf("%02x", (char *)v[0]);
```

- 1) de
- 2) ad
- 3) be
- 4) ef
- 5) 00

ANSWER:

## 3) What is the output of the following code snippet?

```
printf("%d", abs(-2147483648));
```

- 1) 0
- 2) -2147483648
- 3) 2147483648
- 4) NaN
- 5) depending on 32/64-bit arch

ANSWER:

#### 4) What does the expression, $1 > 0$ , evaluate to (on 64-bit)

- 1) 0
- 2) 1
- 3) NaN
- 4) -1
- 5) depending on operating systems

ANSWER:

#### 5) Which value does this code snippet print in x86 (32-bit)?

```
printf("char=%d, int=%d, long=%d", \
      sizeof(char), sizeof(int), sizeof(long));
```

- 1) char=4, int=4, long=4
- 2) char=1, int=4, long=4
- 3) char=1, int=4, long=8
- 4) char=1, int=8, long=8
- 5) compilation error

ANSWER:

#### 6) Which value does this code snippet print?

```
unsigned int i = 0;
printf("%u", i--);
```

- 1) 0
- 2) -1
- 3) 65535

- 4) 4294967295
- 5) compilation error

ANSWER:

## 7) What is the output of the following program?

```
int main ()
{
    int i, j, *p, *q;
    p = &i;
    q = &j;
    *p = 5;
    *q = *p + i;
    printf("i = %d, j = %d\n", i, j);
    return 0;
}
```

- 1) i = 5, j = 10
- 2) i = 5, j = 5
- 3) i = 10, j = 5
- 3) i = 10, j = 10
- 4) Nothing. The program will most likely crash.

ANSWER:

## 8) What's the value of NULL?

- 1) 0xffffffff
- 2) 0x11111111
- 3) 0xf5f5f5f5
- 4) 0x00000000
- 5) 0x00000001

ANSWER:

## 9) What is the output of the following program (32-bit)?

```
main() {
    int x[5];
    printf("1 = %p\n", x);
    printf("2 = %p\n", x+1);
    printf("3 = %p\n", &x);
}
```

```
printf("4 = %p\n", &x+1);  
return 0;  
}
```

(assuming the first printf results in the follow string)  
"1 = 0x7fffdfbf7f00"

ANSWER: 2 = \

ANSWER: 3 = \

ANSWER: 4 =

## 10) Where does the string, "hello world", reside?

```
main() {  
    const char *str = "hello world";  
    printf("%s\n", str);  
}
```

- 1) .text section
- 2) .data section
- 3) .bss section
- 4) stack
- 5) heap

ANSWER:

## 11) Where does the variable 'str' reside?

```
main() {  
    const char *str = "hello world";  
    printf("%s\n", str);  
}
```

- 1) .text section
- 2) .data section
- 3) .bss section
- 4) stack
- 5) heap

ANSWER:

## 12) Where does the function 'main' locate?

```
main() {  
    const char *str = "hello world";  
    printf("%s\n", str);  
}
```

- 1) .text section
- 2) .data section
- 3) .bss section
- 4) stack
- 5) heap

ANSWER:

## 13) Where does the arga locate relative to func's ebp (32-bit)?

```
func(arga, argb, argc, Argo);
```

- 1) ebp - 4
- 2) ebp + 0
- 3) ebp + 4
- 4) ebp + 8
- 5) ebp + 12

ANSWER:

## 14) What is the output of the following program?

```
#define PTXSHIFT 12  
#define PTX(va) (((uint)(va) >> PTXSHIFT) & 0x3FF)  
  
printf("0x%x", PTX(0x12345678))
```

- 1) 0x123
- 2) 0x4561
- 3) 0x234

4) 0x2345

5) 0x345

ANSWER:

### 15) What is the correct usage of this macro?

```
#define ASSERT(a, b) do { switch (0) case 0: case (a): ; } while
```

1) ASSERT(1, 2)

2) ASSERT(1, 2);

3) ASSERT(1 == 2, "error: should be equal");

4) ASSERT(1 != 2, "error: should be equal");

5) ASSERT("error: should be equal", 1 != 2);

ANSWER: