C – More Malloc Free Quiz questions

**Question #0**

To allocate enough space for an array of 10 integers (on a 64bit, Linux machine), I can use:

malloc(10 \* sizeof(int))

**Question #1**

If I want to copy the string “Best School” into a new space in memory, I can use this statement to reserve enough space for it (select all valid answers):

malloc(sizeof(“Best School”))

malloc(12)

malloc(strlen(“Best School”) + 1)

**Question #2**

malloc returns a pointer

True

**Question #3**

malloc returns an address

True

**Question #4**

What is wrong with this code:

int cp(void)

{

char \*s;

s = malloc(12);

strcpy(s, "Best School");

return (0);

}

There is no comment

malloc can fail so we should check its return value all the time before using the pointers returned by the function.

**Question #5**

You can do this:

free("Best School");

No

**Question #6**

You can do this:

char str[] = "Best School";

free (str);

No

**Question #7**

You can do this:

char \*s;

s = strdup("Best School");

if (s != NULL)

{

free(s);

}

Yes

**Question #8**

The memory space reserved when calling malloc is on:

The heap

**Question #9**

What will you see on the terminal?

int main(void)

{

int \*ptr;

\*ptr = 98;

printf("%d\n", \*ptr);

return (0);

}

Segmentation Fault