Started on	Friday, 27 November 2020, 3:35 AM
State	Finished
Completed on	Friday, 27 November 2020, 3:46 AM
Time taken	11 mins 19 secs
Marks	14.00/15.00
Grade	**93.33** out of 100.00

Question 1

Correct

Mark 1.00 out of 1.00

What happens when a block of memory is freed twice, such as in the following block of code?

```
int *arr = malloc(20 * sizeof(int));
free(arr);
free(arr);
```

Select one:

- Nothing, since the memory associated with `arr` has already been freed. cross out
- The second `free` will likely cause a program fault. ✔ cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: The second `free` will likely cause a program fault.

Question 2

Correct

Mark 1.00 out of 1.00

Suppose we have two variables with the same `struct` type, called `s1` and `s2` declared as follows:

```
struct s {
  int a;
  int b;
}
struct s s1, s2;
```

Then, we can copy `s1` into `s2` by simply saying `s2 = s1`.

Select one:

- True ✔ cross out
- False. A `struct` must be copied using something like `memcpy` (similar to arrays) cross out
- False. A `struct` must be copied element-wise. For instance, `s2.a = s1.a, s2.b = s1.b` cross out
- False. A `struct` cannot be copied to another one. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: True

Question 3

Correct

Mark 1.00 out of 1.00

Consider the following declaration:

```
struct {
  char str[5];
  union {
    int y;
    long z;
  } u;
} t;
```

Assume that objects of the type char, int and long occupy 1 bytes, 4 bytes and 8 bytes, respectively. What is the memory requirement for variable t?

Select all that apply:

- [ ] = 10 bytes cross out
- [x] >= 13 bytes ✔ cross out
- [ ] = 17 bytes cross out
- [x] = 16 bytes ✔ cross out

Your answer is correct.

Click "Next page" to continue

The correct answers are: >= 13 bytes, = 16 bytes

Question 4

Correct

Mark 1.00 out of 1.00

How many bytes will an instance of the following `union` occupy?

```
union {
  char a;
  int b;
  long int c;
} u;
```

Select one:

- ( ) sizeof(char) cross out
- ( ) sizeof(int) cross out
- (•) sizeof(long int) ✔ cross out
- ( ) sizeof(char) + sizeof(int) + sizeof(long int) cross out
- ( ) The union definition is invalid. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: sizeof(long int)

Question 5

Correct

Mark 1.00 out of 1.00

How many bytes will an instance of the following `union` occupy?

```
union {
  int b;
  long int c;
} u;
```

Select one:

- ( ) sizeof(int) cross out
- (•) sizeof(long int) ✔ cross out
- ( ) sizeof(int) + sizeof(long int) cross out
- ( ) The union definition is invalid. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: sizeof(long int)

Question 6

Correct

Mark 1.00 out of 1.00

Suppose we have the following program structure.

In main.c:

```
// main.c
#include "a.h"
#include "b.h"
...
```

In a.h:

```
// a.h
#include "b.h"
#define N 100
```

And in b.h:

```
// b.h
struct b {
  int c;
  int d;
};
```

In which file(s) is a header guard (#ifndef ... #endif) required for compilation?

Select one:

- In main.c cross out
- In a.h cross out
- In b.h ✔ cross out
- In a.h and b.h cross out
- All 3 files cross out
- No header guard is required. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: In b.h

Question 7

Correct

Mark 1.00 out of 1.00

Suppose we have the following makefile:

```
main: main.o
  gcc -Wall -std=c99 main.o a.o b.o

a.o: a.c a.h
  gcc -Wall -std=c99 -c a.c

b.o: b.c b.h
  gcc -Wall -std=c99 -c b.c
```

Now, suppose we `make`, and then edit `b.h`, and then `make` again. Which of the rules in the makefile will be run?

Select one:

- main cross out
- a.o cross out
- b.o cross out
- main and a.o cross out
- main and b.o cross out
- All three cross out
- None ✔ cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: None

Question 8

Correct

Mark 1.00 out of 1.00

Make keeps track of when files were last compiled and only recompiles those target files for which the dependency files were changed since make was last executed.

Select one:

- True ✔ cross out
- False cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: True

Question 9

Correct

Mark 1.00 out of 1.00

What file will the following compilation line produce?

```
gcc -Wall -std=c99 -c main.c
```

Select one:

- a.out cross out
- main (an executable) cross out
- main.o ✔ cross out
- No file will be produced. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: main.o

Question 10

Correct

Mark 1.00 out of 1.00

To run the make utility, what file must exist in the directory where you are compiling?

Select all that apply:

- Makefile ✔ cross out
- makefile ✔ cross out
- Make cross out
- make cross out
- Readme cross out
- readme cross out

Your answer is correct.

Click "Next page" to continue

The correct answers are: Makefile, makefile

Question 11

Correct

Mark 1.00 out of 1.00

When used as function parameters, pointers are passed by value.

Select one:

- True ✔ cross out
- False cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: True

Question 12

Correct

Mark 1.00 out of 1.00

Which of the following function prototypes return a pointer to an `int`?

Select one:

- `int f(int *a);` cross out
- `int &f(int a);` cross out
- `int *f(int *a);` ✔ cross out
- Pointers are invalid return types for functions. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: `int *f(int *a);`

Question 13

Correct

Mark 1.00 out of 1.00

What happens when `malloc` cannot find a large enough block of memory to allocate?

Select one:

- `malloc` will find a block of memory that hasn't been accessed for a long time, and reallocate that for this purpose. cross out
- The program will crash. cross out
- `malloc` will return the largest block of memory available. cross out
- `malloc` will return a null pointer. ✔ cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: `malloc` will return a null pointer.

Question 14

Incorrect

Mark 0.00 out of 1.00

Consider the following code fragment:

```
int *arr = malloc(20 * sizeof(int));
arr = realloc(arr, 0);
```

What can we say about `arr` after this code runs?

Select one:

- `arr` is an `int` array that can hold 20 elements cross out
- `arr` is an `int` array that can hold `20 * sizeof(int)` elements cross out
- The behavior is implementation defined. cross out
- The call to `realloc` will fail. cross out
- The code fragment causes undefined behaviour. ✖ cross out

Your answer is incorrect.

Click "Next page" to continue

The correct answer is: The behavior is implementation defined.

Question 15

Correct

Mark 1.00 out of 1.00

Consider the following code fragment:

```
int *arr = NULL;
arr = realloc(arr, 20 * sizeof(int));
```

What can we say about `arr` after this code runs?

Select one:

- `arr` is an `int` array that can hold 20 elements ✔ cross out
- `arr` is an `int` array that can hold 30 elements cross out
- `arr` is an `int` array that can hold `20 * sizeof(int)` elements cross out
- The code fragment causes undefined behaviour. cross out

Your answer is correct.

Click "Next page" to continue

The correct answer is: `arr` is an `int` array that can hold 20 elements

◄ Practice Quiz #10 (up to Lecture 27/Chap 17)

Jump to...

Lab #11 D03 submission page ►