COSC1076 | Advanced Programming Techniques

Revision Questions | Week 04 | Answers

1. (a) Reference to a pointer to an integer
   (b) (solution not provided)
   (c) `-5`. `intRef` reference the pointer of `ptr`, which in turn is a pointer to `value`. So when `intRef` is dereferenced and set to the value -5, the variable `value` is modified (as is where `ptr` points to).

2.
```
char array[length];
char* array;
```

3.
```
char array[rows][cols];
car** array;
```

4. (a) Algorithm, Assuming this is for part b
      i. Create memory for an array of pointers to character
      ii. For each cell of the above array, create memory for an array of characters
   (b) (solution withheld)
5. To ensure any resources used by the class during its lifetime are correctly cleaned up and deallocated.
6. The automated memory management uses the program stack, programmer managed memory uses the heap
7. (a) The file `A.h` is included twice (because it is also included in `B.h`)
   (b) Changes are:

A.h
```
#ifndef A_HEADER
#define A_HEADER

int foo();

#endif // A_HEADER
```

B.h
```
#ifndef B_HEADER
#define B_HEADER

#include "A.h"

int bar();

#endif // B_HEADER
```

8. (a) On the heap
   (b) The `main` function. This is because it deletes the pointer, and the deconstructor of the `Ownership` class does not.
   (c) Changes are:

Ownership.cpp
```
#include "Ownership.h"

Ownership::Ownership(int* ptr) {
    this->ptr = ptr;
```

```
}

Ownership::~Ownership() {
    delete ptr;
}
```

main.cpp

```
#include "Ownership.h"

#define EXIT_SUCCESS    0

int main(void) {
    int* ptr = new int(-1);
    Ownership* owner = new Ownership(ptr);

    delete owner;

    return EXIT_SUCCESS;
}
```

9. Commands are:

```
g++ -Wall -Werror -std=c++14 -O -c Ownership.cpp
g++ -Wall -Werror -std=c++14 -O -c main.cpp
g++ -Wall -Werror -std=c++14 -O -o program Ownership.o main.o
```