C Test

Question 1

Please refer to `Question1` project within the `JazzNetwork.sln` solution. The source code is in `Question1\Question1.cpp` file.

It is suggested to run this solution in Visual Studio 2017, with `Ctrl + F5` command, to see the output.

The solution prints out the following:

- Original linked list
- Original linked list after removal of even valued nodes
- New linked list containing the removed even valued nodes

Question 2

```
1  #include <stdio.h>
2  #include <string.h>
3
4  char *f(int m){
5    char buf[6];
6    int x;
7    if (m == 1 && x--) {
8        strcpy(buf, "AAAAAA");
9        return buf;
10   } else if (m == 2) {
11       char *msg = (char *)malloc(100);
12       strcpy(msg, "BBBBBB");
13       return msg;
14   }
15 }
16
17 int main(int argc, char **argv) {
18   char *m;
19   m = f(argc);
20   putchar(m[0]);
21   return 0;
22 }
```

Several observations of the code are as follows:

- Argument `char **argv` in line 17 is unused.
- Integer `x` in line 6 is undefined. Outcome of `x--` on line 7 is thus undefined.
- `strcpy` function is unsafe. Since `strcpy` has no knowledge of size of destination buffer, it might overrun the destination buffer and corrupt the surrounding memory.

- `strcpy` on line 8 copies a char array of length 7 (including the null character `\0`) into a `buf` variable of size 6 resulting in the stack around the variable `buf` being corrupted.
- The `<stdlib.h>`, where `malloc()` is declared, is not included.
- On line 11, `malloc()` allocates 100 bytes but only 7 bytes are filled with string `"BBBBBB"`. Although the code does not access the extra memory locations, accessing them will return undefined values.
- If `malloc()` fails to allocate the requested block of memory, it returns a null pointer. This case is not handled by the code.
- Return value for function `f` is undefined when m != 1 and m != 2.

Question 3

Please refer to `Question3` project within the `JazzNetwork.sln` solution. The source code is in `Question3\Question3.cpp` file.

It is suggested to run this solution in Visual Studio 2017, with `Ctrl + F5` command, to see the output.