C Programming

1. Given:

```
struct node_t
{
   unsigned v;
   struct node_t* next;
};
```

A singly linked list of nodes is referred to by a pointer to the head node (the head pointer will be `NULL` in the case of an empty list).

Write a function:

```
struct node_t* even_nodes(struct node_t**)
```

which is passed a pointer to the head pointer of a list. It should remove all nodes with even values from the indicated list, putting them in a new list, which is returned.

2. Find the instances of undefined behaviour in the following code:

```
#include <stdio.h>
#include <string.h>

char *f(int m)
{
  char buf[6];
  int x;

  if (m == 1 && x--)
  {
    strcpy(buf, "AAAAAA");
    return buf;
  }
  else if (m == 2)
  {
    char *msg = (char *)malloc(100);
    strcpy(msg, "BBBBBB");
    return msg;
  }
}

int main(int argc, char **argv)
{
  char *m;
  m = f(argc);
  putchar(m[0]);
  return 0;
}
```

3. Write a C function `reverse_bytes()` taking two parameters and returning no result. The first parameter is a pointer to a buffer containing n contiguous bytes (each of type `unsigned char`), and the second is a count of the number of bytes. The function should reverse the order of the bits in the n contiguous bytes, which is seen as a bitstring of length 8n. For example, the first bit of the first byte should be swapped with last bit of the last byte.