

CS 250
Homework 2-2

1. Write a function that takes an `unsigned int` as a parameter, and returns the number of one's in its binary representation. Assume for this question that an `unsigned int` contains 32 bits. (Hint: Use the bitwise-and operation to mask all but the right-most bit of the parameter, and use the right-shift operation to discard the right-most bit after you have tested it.)
2. Write a function `compress()` that takes a C-string as a parameter, and compresses it so that all the non-alphabetic characters have been removed and the remaining characters have been pushed towards the beginning of the string. For example, if your function is passed the C-string "This string is 2 odd!", it should change it to the C-string "Thisstringisodd".

3. In class we wrote an implementation of the standard library function `atoi()` that converts a C-string that contains only digits (with an optional leading '-') to the `int` it represents. Write a function that converts an `int` to the C-string that represents it. Don't forget to handle the case of a negative value. Your function should have the following prototype:

```
void itoa (int i, char s[]);
```

Assume that the array that is passed to the function is large enough to hold the C-string.

4. Write a function called `substr()` that determines if the second parameter of the function is a substring of the first parameter. Your function should return true if it is, and false otherwise. Your function should have the following prototype:

```
int substr (const char s[], const char t[]);
```

Do not use any functions in `string.h` (p. 249) except `strlen()`.