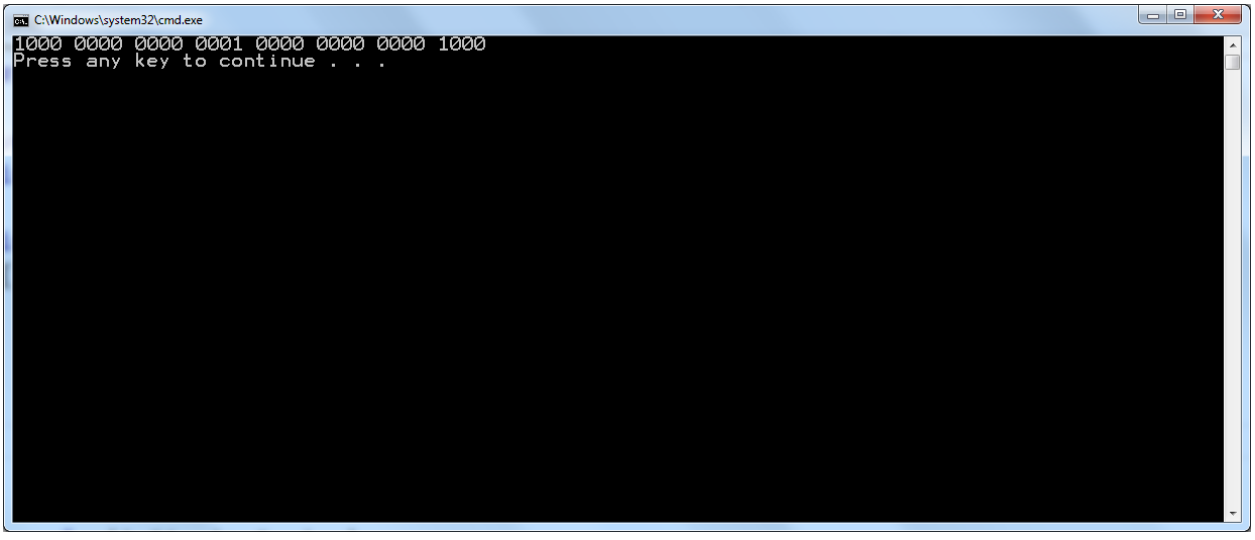
This daily will allow you to practice more with the bit wise operators and shifts. Consider the following main program:

```
#include <stdio.h>

void set_flag(int* flag_holder, int flag_position);
int check_flag(int flag_holder, int flag_position);

int main(int argc, char* argv[])
{
	int flag_holder = 0;
	int i;

	set_flag(&flag_holder, 3);
	set_flag(&flag_holder, 16);
	set_flag(&flag_holder, 31);

	for(i=31; i>=0; i--)
	{
		printf("%d", check_flag(flag_holder, i));
		if(i%4 == 0)
		{
			printf(" ");
		}
	}
	printf("\n");
	return 0;
}
```

Write the code for the definition of set_flag and check_flag so that the output of your program looks like the following:



You can think of the set_flag function as taking an integer and making sure that the n^{th} bit is a 1. The check_flag function simply returns an integer that is zero when the n^{th} bit is zero and 1 when it is 1. You may find the shifting

operators “<<”, and “>>” helpful as well as the bitwise operations & and |. If you find yourself using multiplication or division in your solution then you are doing it wrong.

At the top of your code you should have a comment section that has the following format:

```
/*********************************************************
       Author: <your name>
       Date: <Today's date>
       Effort: <Time you spent on this project>
       Purpose: <Purpose of this assignment in your own words>
*********************************************************/
```