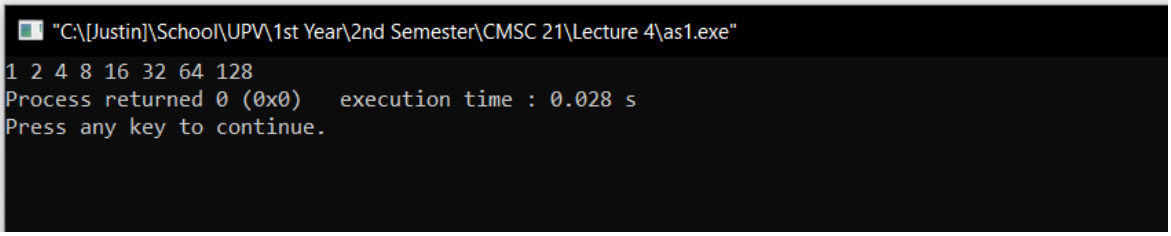
1. The output of the program is “1 2 4 8 16 32 64 128”

Code and Output:

```c
#include <stdio.h>
int main(void)
{
 int i;

 i = 1;

 while (i <= 128) {
 printf("%d ", i);
 i *= 2;
 }

 return 0;
}
```



2. All three statements are equivalent when i as a starting variable is less than 10. However, when i as a starting variable is greater than or equal to 10, then only statements A and B are equivalent because they will not execute unless i is less than 10. A and B will have no output.

   Since C is a do-while statement, it will first execute the loop body, produce an output, and then evaluate i. Therefore, C is not equivalent to A and B.

   Example code and output showing equivalence between A, B, C:

Statement A:

as1.c | as2(Statement A).c | as2(Statement B).c | as2(Statement C).c | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

i = 1;

while (i<10){

  printf("%d ", i);
  i++;
}

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement A).exe"
1 2 3 4 5 6 7 8 9
Process returned 0 (0x0)   execution time : 0.027 s
Press any key to continue.
```

Statement B:

as1.c | as2(Statement A).c | as2(Statement B).c | as2(Statement C).c | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

for (i=1;i<10;i++){

  printf("%d ", i);

}

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement B).exe"
1 2 3 4 5 6 7 8 9
Process returned 0 (0x0)   execution time : 0.026 s
Press any key to continue.
```

Statement C:

as1.c | as2(Statement A).c | as2(Statement B).c | **as2(Statement C).c** | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

i = 1;

do {

  printf("%d ", i);
  i++;
}while (i<10);

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement C).exe"
1 2 3 4 5 6 7 8 9
Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

Example code and output showing C as not equivalent to A and B.

Statement A and B:

as1.c | **as2(Statement A).c** | *as2(Statement B).c | as2(Statement C).c | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

i = 10;

while (i<10){

  printf("%d ", i);
  i++;
}

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement A).exe"

Process returned 0 (0x0)   execution time : 0.027 s
Press any key to continue.
```

as1.c | as2(Statement A).c | **as2(Statement B).c** | as2(Statement C).c | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

for (i=10;i<10;i++){

  printf("%d ", i);

}

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement B).exe"

Process returned 0 (0x0)   execution time : 0.026 s
Press any key to continue.
```

Statement C:

as1.c | as2(Statement A).c | as2(Statement B).c | **as2(Statement C).c** | as3.c | as4.c | as5.c

```c
#include <stdio.h>

int main(){

int i;

i = 10;

do {

  printf("%d ", i);
  i++;
}while (i<10);

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as2(Statement C).exe"
10
Process returned 0 (0x0)   execution time : 0.026 s
Press any key to continue.
```

3.

Code:

```c
#include <stdio.h>

int main(void) {

int i;

for (i = 1; i <= 128; i *= 2){
 printf("%d ", i);
}

return 0;

}
```

Sample output:

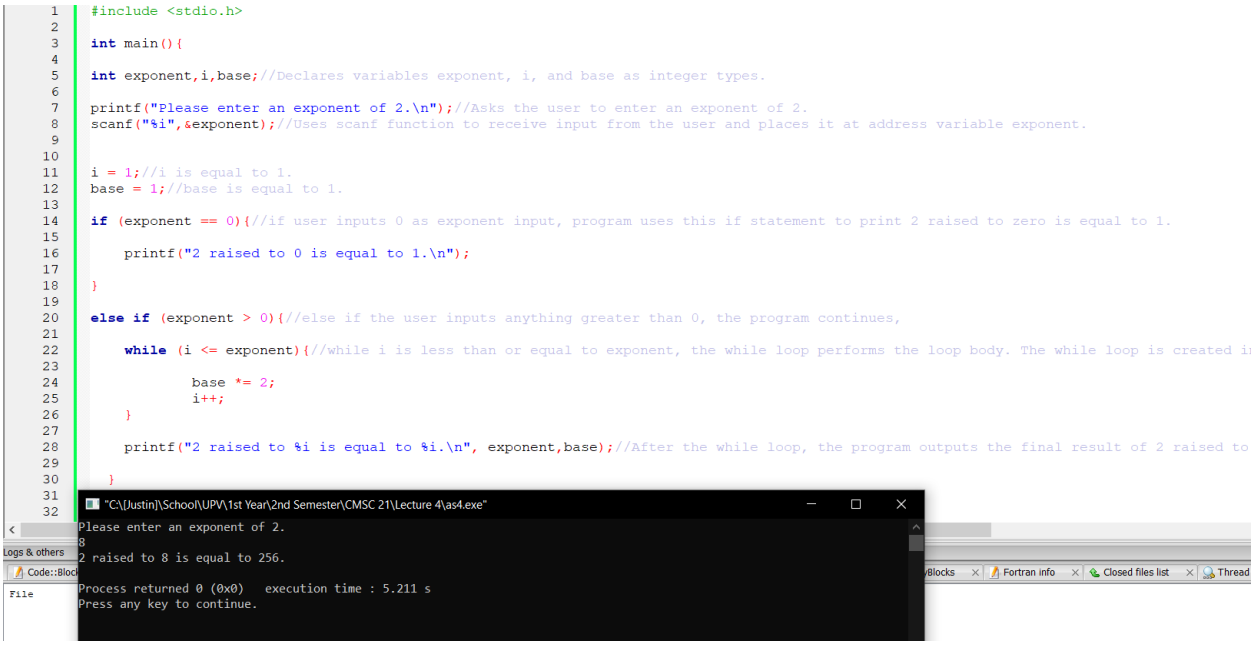
```c
#include <stdio.h>

int main(void) {

int i;

for (i = 1; i <= 128; i *= 2){
 printf("%d ", i);
}

return 0;

}
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as3.exe"
1 2 4 8 16 32 64 128
Process returned 0 (0x0)   execution time : 0.028 s
Press any key to continue.
```

4. Code:

```c
#include <stdio.h>

int main(){

int exponent,i,base;//Declares variables exponent, i, and base as integer types.

printf("Please enter an exponent of 2.\n");//Asks the user to enter an exponent of 2.
scanf("%i",&exponent);//Uses scanf function to receive input from the user and places it at address variable exponent.


i = 1;//i is equal to 1.
base = 1;//base is equal to 1.

if (exponent == 0){//if user inputs 0 as exponent input, program uses this if statement to print 2 raised to zero is equal to 1.

    printf("2 raised to 0 is equal to 1.\n");

}

else if (exponent > 0){//else if the user inputs anything greater than 0, the program continues,

    while (i <= exponent){//while i is less than or equal to exponent, the while loop performs the loop body. The while loop is created in a w

            base *= 2;
            i++;
    }

    printf("2 raised to %i is equal to %i.\n", exponent,base);//After the while loop, the program outputs the final result of 2 raised to what

  }

```

Sample output:



5. Code:

```c
#include <stdio.h>

int main(){

int days,starting_day,test,test1,i,j,row_count;//Declares days,starting_days,test,test1,i,j,and row_count as integer type variables.

//variables for the program.
test = 0;//used for 1st test of user input
test1 = 0;//used for 2nd test of user input.
i = 1;//used for printing the calendar.
j = 1;//used for printing the space as indicated by starting day.
row_count = 0;//used to print a new line every time row reaches 7 spaces.


while (test < 1){//while loop for checking if user input is between 28-31. while test is less than 1, the loop body repeats.
printf("Enter number of days in month: ");//asks the user to input number of days in month.
scanf("%d", &days);//uses scanf function to receive input and place it into address variable days.

if (days > 31 || days < 28){//if days is greater than 31 or less than 28, the program informs the user their input is not valid and will add 0 to test.

    printf("Error, entered days not valid. Please enter 28-31 for days in month.\n");
    test += 0;

        }

else if (days > 27 && days < 32){//else if days is between 28-31, then the program adds 1 to test so that the program can continue.
    test ++;

}

}
while (test1 < 1){//while loop for checking if user input is between 1-7 for the starting day of the week. while test1 is less than 1, the loop body repeats.

printf("\nEnter the starting day of the week (1= Sun, 7= Sat): ");//asks the user to input the starting day of the week.
scanf("%d",&starting_day);//uses scanf function to receive input and places it at address variable starting_day.

if (starting_day > 7 || starting_day < 1){//if starting day is less than 1 or greater than 7, the program informs the user that their input is not valid and w

    printf("Error, starting day not valid. Please enter a number from 1-7 for starting day.\n");
    test1 += 0;

        }

else if (starting_day >= 1 && starting_day <= 7){//else if starting_day is between 1-7, the program adds 1 to test1 so it can continue.
    test1 ++;



}

}
while (i <= days){//while i is less than or equal to days,

        while (j < starting_day) {//while j is less than starting_day,the program prints 2 spaces then increments row_count and j by 1.

        printf("   ");
        row_count++;
        j++;
        }

   if (i < 10){//if i is a single digit number, it prints i with one space before and after it then increments i and row_count by 1.
    printf(" %d ",i);
    i++;
    row_count++;
    }

   else if (i > 9){//else if i is a two digit number, it prints i followed by only one space after it, then increments i and row_count by 1.
   printf("%d ",i);
   i++;
   row_count++;
   }

   if (row_count % 7 == 0){//if row_count modulo 7 is equal to 0, meaning that the spaces filling one row is 7, the program prints a new line.
        printf("\n");
   }

}
```

Sample output:

```
55
56          while (j < starting_day) {//while j is less than starting_day,the program prints 2 spaces then increments row_count and j by 1.
57
58          printf("   ");
59          row_count++;
60          j++;
61          }
62
63      if (i < 10){//if i is a single digit num
64      printf(" %d ",i);
65      i++;
66      row_count++;
67      }
68
69      else if (i > 9){//else if i is a two dig
70      printf("%d ",i);
71      i++;
72      row_count++;
73      }
74
75      if (row_count % 7 == 0){//if row_count m
76          printf("\n");
77      }
78
79  }
80
81
82
83
84  }
85
86
```

```
"C:\[Justin]\School\UPV\1st Year\2nd Semester\CMSC 21\Lecture 4\as5.exe"
Enter number of days in month: 31

Enter the starting day of the week (1= Sun, 7= Sat): 8
Error, starting day not valid. Please enter a number from 1-7 for starting day.

Enter the starting day of the week (1= Sun, 7= Sat): 5
          1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31

Process returned 0 (0x0)   execution time : 7.475 s
Press any key to continue.
```