**Lara Eunice A. Pacumio**

**2011-27228**

**CS11 THCW**

**ME7**

1. Describe what function print is doing. How does it differ from printf? Why is this function void? Why not int or char instead?

The function print prints a certain character (k) a certain number (count) of times.

Its difference with printf is that printf only prints a character (or number) once while in print, the character is printed count times.

A function is void when it does not return a value or a character. When the function print is called, the function itself already prints the value or character that needs to be printed. It does not need to return anything to the main function so it is void.

It is not int or char because it doesn’t need to return an integer or a character. However, it may still be made into an int or char function if the programmer wants to but it is not necessary because the value that will be returned will not be used.

2. Describe what function fun is doing. Describe the process of each loop.

Function fun prints diamonds with width of input. If it is executed only once, it only prints a line of diamonds but since, in function main, function fun is in a loop, it is executed many times while the number of diamonds that should be printed is increased(and then decreased once the correct width is obtained). At the end of the program, a large diamond is formed, made up of smaller diamonds.

The first loop that will be encountered with the condition “while (str\_count<=a)” first prints a number of spaces so that when the stars are printed, it is in the right position. The stars are printed using another loop.

The loop inside the first loop prints the stars. It is in a loop so that the spaces and stars are printed b number of times. It needs to be b times so that many diamonds will be formed in a line once the function fun is done. Thus, when the whole program is done, a larger diamond will be formed.

After printing the stars, the program prints a new line then adds 1 to the value of the number of stars to be printed and decreases the number of spaces. The loop is executed until the number of stars that was printed is the correct width of the diamond. At the end, the upper half of the line of diamonds should have been printed.

The next loop that will be encountered is very similar to the first except that instead of increasing the number of stars, it decreases it and instead of decreasing the spaces, it increases. The loop inside this still does the same thing as the loop inside the first one. This loop prints the lower half of the line of diamonds.

Once the function is done, there should be a line of diamonds.

3. Identify what a, b and c are and describe their purpose in the function.

‘a’ is the width of the smaller diamond being formed.

‘b’ is the number of diamonds that should be printed on each line.

‘c’ is the number of spaces(considering that one unit of space is the width of the smaller diamonds) that should be printed before printing the diamonds. To get the exact number of spaces, the values of a and c are multiplied.

4. Why do we have to copy the value of b? Why not use b directly instead?

If b was used directly, its value will change and end up as 0. B is the number of times the loop is executed but if it was used directly, not copy\_b, then once the value of b becomes 0, it will not execute the loop again because it is already 0. If that happens, the output will be very different from what it should be.

5. What does this statement do and why do we need to do this?

This statement prints spaces before the diamond. If this wasn’t present in the program then the output will not be a larger diamond composed of smaller diamonds but, instead, the smaller diamonds form a triangle where they are aligned at the left.

6. What is the value of str\_count after the loop above? Explain how it arrives to that value.

At the end of the loop, the value of str\_count is a+2 because at the start its value is 1 but every time the loop is executed, it increases by 2. The loop is executed for the last time once str\_count = a but at the loop, str\_count is still increased by 2 so at the end, str\_count = a+2.

7. What is the value of spc\_count after the loop above? Explain how it arrives to that value.

At the end of the loop, the value of spc\_count is a/2 (this is integer division) and it is decreased by 1 every time the loop is executed. When the loop is executed for the last time, its value is 0 but at the end of the loop, it is still decreased by 1 so the value is -1.

7. Why do we have to assign a-2 to str\_count?

From the previous loop, the line with a number of star is already printed. The next to be printed is a line with a-2 number of starts. If this value wasn’t assigned, the previous value of str\_count would have been used which is a+2 which gives a very different output.

8. Why do we have to assign 1 to spc\_count?

The last number of spaces that was printed was zero. The next that should be printed is 1. If one wasn’t assigned to the value, the previous value would have been used which is -1 which will make an error to the output.

9-10. Write a pseudocode for print and fun functions. Describe the input, process and output.

Void print(char k, int count)

while(count>0)

printf(“%c”,k)

count--;

In function print, the parameters is one character that is to be printed and an integer that is to be the number of times the character will be printed.

The command printf is inside a loop that repeats count number of times so that the character is printed the same number of times.

Void fun(int a, int b, int c)

pre\_spc = a\*c

str\_cnt = 1

spc\_cnt = a/2

while(str\_cnt<=a)

copy\_b = b

print(‘ ‘,pre\_spc)

while(copy\_b>0)

print(‘ ‘,spc\_cnt)

print(‘\*’,str\_cnt)

print(‘ ‘,spc\_cnt)

copy\_b—

print(“\n”,1)

str\_cnt+=2

spc\_cnt--

str\_cnt = a-2

spc\_cnt = 1

while(str\_cnt>=1)

copy\_b = b

print(‘ ‘,pre\_spc)

while(copy\_b>0)

print(‘ ‘,spc\_cnt)

print(‘\*’,str\_cnt)

print(‘ ‘,spc\_cnt)

copy\_b--

print(“\n”,1)

str\_cnt-=2

spc\_cnt++

The parameters are a, b, and c which are the width of the small diamonds, the number of diamonds that should be printed, and the number of spaces, respectively.

On the loop, first, the spaces are printed so that the diamonds will be placed in the right position. On the loop inside it, the space and the stars are printed so that a diamonds will be formed.

The first loop prints the upper half of the set of diamonds and the second loop prints the lower half.